P 1877/71

6 1971

informatyka

SPIS TREŚCI

Jerzy Bogdanowicz, Jerzy Sukiennik — „SAROP — system automatycznego rozliczania płac i kosztów robocizny w Stoczni im. Komuny Paryskiej w Gdyni 1

Zbigniew Ładoś — „Uruchamianie nowych programów na komputerze NCR 315" 6

Antoni Nadolski — „System eksploatacji programów na komputerze NCR 315" 10

Jerzy Wojciechowski, Ludmiła Zawadzka — „APT — system automatycznego programowania technologii na obrabiarki sterowane liczbowo" . . . 14

Tomasz Pawlak — „COMPUTER 70 — Światowe spotkanie producentów i użytkowników maszyn cyfrowych" — Londyn, 5—9.X.1970 18

Andrzej Targowski — „Dynamika skali i kierunków rozwoju informatyki w USA część I 21

PORZĄDKOWANIE TERMINOLOGII MASZYN MATEMATYCZNYCH

„Projekt normy pojęć podstawowych, część II — Programowanie" . . 22

Z KRAJU i ze ŚWIATA . . . . . IV str. okł.

Z życia SEP 9

PRZEGLĄD WYDAWNICTW . . . . . III, IV str. skrzydełka

WYDAWNICTWA CZASOPISM TECHNICZNYCH NOT
Warszawa
Czackiego 3/5

KOLEGIUM REDAKCYJNE
Redaktor naczelny prof. dr Leon ŁUKASZEWICZ

Doc. dr hab. in. Konrad FIAŁKOWSKI (zast. redaktora naczelnego), Władysław KLEPACZ, dr Antoni MAZURKIEWICZ, in. Dorota PRAWDZIC (zast. redaktora naczelnego), dr inż. Andrzej TARGOWSKI

Sekretarz Redakcji mgr Wanda KAČER Redaktor techniczny Leszek ZAGAŃSKI

RADA PROGRAMOWA

Mgr inż. Jan Bursche, mgr inż. Henryk Chyrek, (wiceprzewodniczący) mgr inż. Ryszard Dąbrówka, mgr inż. Bolesław Gliksman, mgr inż. Józef Knysz, proof. dr Leon Łukaszewicz, mgr inż. Jan Matejak, prof. dr Tadeusz Peche (przewodniczący), mgr inż. Jerzy Trybulski (wiceprzewodniczący), dr Tadeusz Walczak, mgr Kazimierz Wasilewski, mgr Waldemar Wiśniewski (sekretarz), mgr Stefan Wojciechowski, dr inż. Hentryk Woźniacki, mgr inż. Jan Zdzisław Żydowo

Redakcja: Warszawa, ul. Emilii Plater 20 m. 15, tel. 21-13-91. Zastępca redaktora naczelnego tel. 28-37-29

Zakład Kolportażu WCT NOT, Warszawa, ul. Mazowiecka 12

Zakł. Graf. „Tamka". Z. 2. Zam. 256. Papier powlekany V kl. 80 g. Obj. 3 ark. druk. Nakład 3000. U-47.

Cena egzemplarza zł 8.— INDEKS 36707 Prenumerata roczna zł 96.—

P. 1877/71

Informatyka

dawniej Maszyny Matematyczne

zastosowania w gospodarce, technice i nauce

Nr 6
MIESIĘCZNIK
1971
ROK VII
Czerwiec

ORGAN KRAJOWEGO BIURA INFORMATYKI I POLSKIEGO KOMITETU AUTOMATYCZNEGO

Dr inż. Jerzy Bogdanowicz ukończył Wydział Budowy Okrętów Politechniki Gdańskiej w roku 1960. Następnie specjalizował się w Leningradzkim Instytucie Budowy Okrętów. Pracował początkowo w Gdańskiej Stoczni Remontowej, a następnie był kierownikiem Zakładu ETO Instytutu Morskiego w Gdańsku oraz kierownikiem ZOPI w Stoczni im. Komuny Paryskiej w Gdyni. Obecnie jest zatrudniony w aparacie KW PZPR w Gdańsku.

Inż. Jerzy Sukiennik ukończył studia w Wyższej Szkole Ekonomicznej w Pradze (CSRS) na Wydziale Mechanizacji i Automatyzacji Zarządzania. Od roku 1967 pracuje w Stoczni im. Komuny Paryskiej w Gdyni, najpierw jako kierownik Działu Projektowania, a obecnie jako kierownik Zakładowego Ośrodka Przetwarzania Informacji.

JERZY BOGDANOWICZ,
Zakładowy Ośrodek
Przetwarzania Informacji
Stoczni im. Komuny Paryskiej
Gdynia
JERZY SUKIENNIK

629.128:658.32:658.53

SAROP — system automatycznego rozliczania płac i kosztów robocizny w Stoczni im. Komuny Paryskiej w Gdyni[1)]

Podano sposób przygotowania (wraz z kontrolą) danych wejściowych na kartach, schemat przetwarzania oraz ogólną charakterystykę programów i wzorów tabulogramów. System automatycznego rozliczania płac i kosztów robocizny, wdrożony w Stoczni im. Komuny Paryskiej w Gdyni, wykorzystuje komputer ICT 1904.

Artykuł ma na celu pokazanie warunków pracy Zakładowego Ośrodka Przetwarzania Informacji Stoczni im. Komuny Paryskiej w Gdyni, realizującego obliczenia częściowo za pomocą własnego parku maszynowego, a częściowo — maszyny cyfrowej zainstalowanej poza Ośrodkiem.

Takie warunki organizacyjne determinują sposób przygotowania, wdrażania i eksploatacji systemu. Rzucają nowe światło na sprawę odpowiedzialności za jakość projektów, programów i instrukcji eksploatacji oraz za jakość tabulogramów wynikowych.

[1)] Stocznię tę — w odróżnieniu od innych stoczni — będziemy w całym artykule nazywać Stocznią z dużej litery

Realizacja tych etapów przez własnych pracowników Ośrodka nakłada pewne dodatkowe obowiązki, ale jest w ostatecznym efekcie bardzo opłacalna. Eksploatacja zautomatyzowanych systemów płac jest procesem bardzo trudnym i odpowiedzialnym. Szczególnie istotnym zadaniem jest wszechstronne sprawdzenie na etapie wdrażania algorytmów zawartych w poszczególnych programach.

I. Charakterystyka Stoczni

Stocznia im. Komuny Paryskiej w Gdyni jest jedną z trzech dużych polskich stoczni produkcyjnych, zgrupowanych w Zjednoczeniu Przemysłu Okrętowego. Stocznia ta jest najnowocześniejszym w Polsce zakładem budowy statków. Posiada suchy dok przeznaczony do budowy dużych statków oraz wyposażona jest w nowoczesne i wydajne urządzenia do budowy kadłubów statków do wielkości ponad 100 000 DWT.

Aktualnie największym statkiem zbudowanym w Stoczni jest masowiec „Manifest Lipcowy" o nośności 55 000 DWT.

Stocznia znana jest szeroko w świecie ze stosowania nowoczesnych metod technologicznych przy budowie kadłubów. Są to:

- połówkowa metoda budowy kadłubów w doku
- spawanie połówek kadłubów dużych statków na wodzie poza niecką suchego doku.

Osiągnięcia te stawiają przed Stocznią ogromne perspektywy rozwoju, czego wyrazem jest zawarcie wielu kontraktów na eksport statków na rynki krajów kapitalistycznych. Stocznia należy do przedsiębiorstw o najbardziej skomplikowanej technologii i organizacji produkcji. Obrazowo można to zilustrować następującymi faktami:

- rocznie buduje się około 20 statków różnych typów — od niewielkich trawlerów rybackich na pochylni do ogromnych masowców w suchym doku. Najmniejszy z tych statków przekracza wartość 100 mln zł, co w warunkach budownictwa lądowego oznacza koszt budowy dużej fabryki
- statki buduje się jednocześnie — kilka lub kilkanaście pływających fabryk z całym zapleczem socjalnym
- budowa statków charakteryzuje się długim cyklem konstrukcyjnego, technologicznego i organizacyjnego przygotowania produkcji oraz długim cyklem właściwego wykonawstwa, montażu stoczni (łącznie cykl kontraktowy wynosi około 3 lat)
- Stocznia pozostaje w związkach kooperacyjnych i dostawczych z ponad 600 przedsiębiorstwami
- na statkach równolegle z wydziałami produkcyjnymi Stoczni wykonuje roboty kilkanaście wyspecjalizowanych przedsiębiorstw spoza Stoczni
- przez Stocznię przechodzi rocznie ogromna masa materiałów i urządzeń, szczególnie materiałów hutniczych (blach i profili).

W tak skomplikowanych warunkach istnieje nieodzowna konieczność stosowania nowoczesnych metod planowania i kontroli przebiegu produkcji oraz nowoczesnych metod ewidencji, przetwarzania i analizy danych do potrzeb zarządzania.

Z tych konieczności doskonale zdaje sobie sprawę Kierownictwo i Zarząd Stoczni: podjęto szereg prób usprawnienia sytuacji. Stocznia jest wyposażona w środki małej, średniej i wielkiej mechanizacji (Stacja MLA ARITMA) oraz od kilku lat korzysta z usług maszyny cyfrowej ICT 1904, zainstalowanej w ZETO Gdynia.

Większe możliwości stosowania elektronicznej techniki obliczeniowej rysują się przed Stocznią po wyposażeniu przemysłu okrętowego w maszynę ICL 4/70 w roku 1971.

Obecne możliwości automatyzacji przetwarzania informacji są ograniczone brakiem wystarczającej ilości czasu maszyny cyfrowej (limit w ZETO Gdynia wynosi około 40 godzin miesięcznie).

Mimo tych ograniczeń, udało się osiągnąć następujące sukcesy:

- wyszkolenie kadry projektantów i programistów EPD
- przygotowanie i wdrożenie dużego systemu EPD SAROP
- wdrożenie kilku systemów planowania operatywnego produkcji na wydziałach kadłubowych oraz kilku małych systemów z różnych dziedzin
- podjęcie prób zastosowania metod optymalizacji przebiegu produkcji.

II. Główne zasady systemu płac w Stoczni im. Komuny Paryskiej

Artykuł niniejszy traktuje szerzej o płacach brutto pracowników fizycznych w Stoczni ze względu na znaczny stopień skomplikowania systemu klasyfikacji i naliczania składników tych płac. Sama liczba składników osobowego funduszu płac (płac podstawowych i uzupełniających) przekracza 50. Biorąc pod uwagę zatrudnienie w Stoczni — około 7000 robotników — powstaje masa operacji obrachunkowych, wykonywanych 2 razy w miesiącu (do zaliczki i do wyrównania) w bardzo krótkim czasie.

Dodatkowo komplikuje ten proces bardzo złożony algorytm rozliczania akordu zespołowego, gdzie zarobek zasadniczy oraz godziny normowane rozliczane są w stosunku do godzin przepracowanych i grup osobistego zaszeregowania każdego robotnika pracującego w zespole. Karta pracy w warunkach Stoczni jest jednocześnie nośnikiem informacji o pracochłonności produkcji wykonanej w danym okresie przez robotnika, zespół mistrzowski, wydział i całą Stocznię, oraz o kosztach robocizny.

III. Dotychczasowy sposób rozliczania płac

Nowy system skorzystał z doświadczeń uzyskanych przy eksploatacji systemu płac, realizowanego za pomocą MLA ARITMA.

Powodem przeniesienia systemu na maszynę cyfrową były następujące okoliczności:

- niewystarczająca przepustowość posiadanego parku maszynowego MLA — 1 tabulator T 320
- ogromne spiętrzenie pracy w stacji MLA, powodowane nierytmicznym spływem dowodów źródłowych
- pewne trudności technologiczne, a mianowicie —

1. Wysokie wymagania w stosunku do jakości kart maszynowych, a szczególnie dokładności wymiarów tych kart i rozmieszczenia otworów perforowanych.

2. Problemy archiwowania kart przez długi okres budowy statków — częste niezgodności zestawień po zakończeniu budowy z zestawieniami miesięcznymi.

3. Niska jakość formy zewnętrznej zestawień wynikowych z powodu braku możliwości tabelowania oraz opisywania nagłówków, sum itp., powodująca znaczne nakłady czasu na „rozszyfrowanie" tabulogramów przez użytkowników.

4. Długi czas sporządzania zestawień — do dnia 15 po zakończeniu miesiąca obrachunkowego.

IV. Przygotowanie danych wejściowych

Podstawowym nośnikiem informacji jest KARTA ROBOCZA specjalnie dostosowana do systemów płac w Stoczni, używana dla trzech podstawowych systemów płac robotników:

- A — akord indywidualny i zespołowy
- B — dniówka limitowana zadaniami
- D — dniówka produkcyjna, pomocniczo-produkcyjna lub zryczałtowana.

W tym ostatnim przypadku stosowany jest również wzór zmniejszony CWD.

W przypadku systemu akordowego stosowane są pewne modyfikacje wzoru karty pracy, spowodowane specyfiką opisywania operacji technologicznych przez niektóre wydziały produkcyjne (np. nadruk stałych członów treści operacji).

Ciekawszym eksperymentem jest również wprowadzenie w Stoczni tzw. akordu zryczałtowanego, gdzie wartość robocizny całkowitej na danej operacji jest sumą robocizny technologicznej i dodatków z tytułu pracy w warunkach szkodliwych i uciążliwych dla zdrowia oraz przy pracach na wysokościach, w warunkach zimowych lub niebezpiecznych.

W tym przypadku odchodzi się od kontroli wykonania normy na każdej operacji (nie specyfikuje się godzin normowanych ani przepracowanych) oraz pozostawia się swobodę podziału zarobku pomiędzy członków danego zespołu (brygady).

Eksperyment ten ma zachęcić robotników do zwiększenia wydajności pracy, co jest dla Stoczni bardzo istotne ze względu na deficyt pracowników w niektórych zawodach (np. malarz — izolator okrętowy).

Dalszym istotnym dokumentem jest specjalny wzór karty zegarowej, dopasowanej zarówno do potrzeb kontroli dyscypliny pracy, jak i do potrzeb ewidencji składników czasu nominalnego robotników.

Karta ta jest jednocześnie ważnym nośnikiem informacji do maszynowego obliczania płac, ponieważ podaje dane o czasie przepracowanym w godzinach nadliczbowych, w nocy oraz dodatku dla brygadzistów.

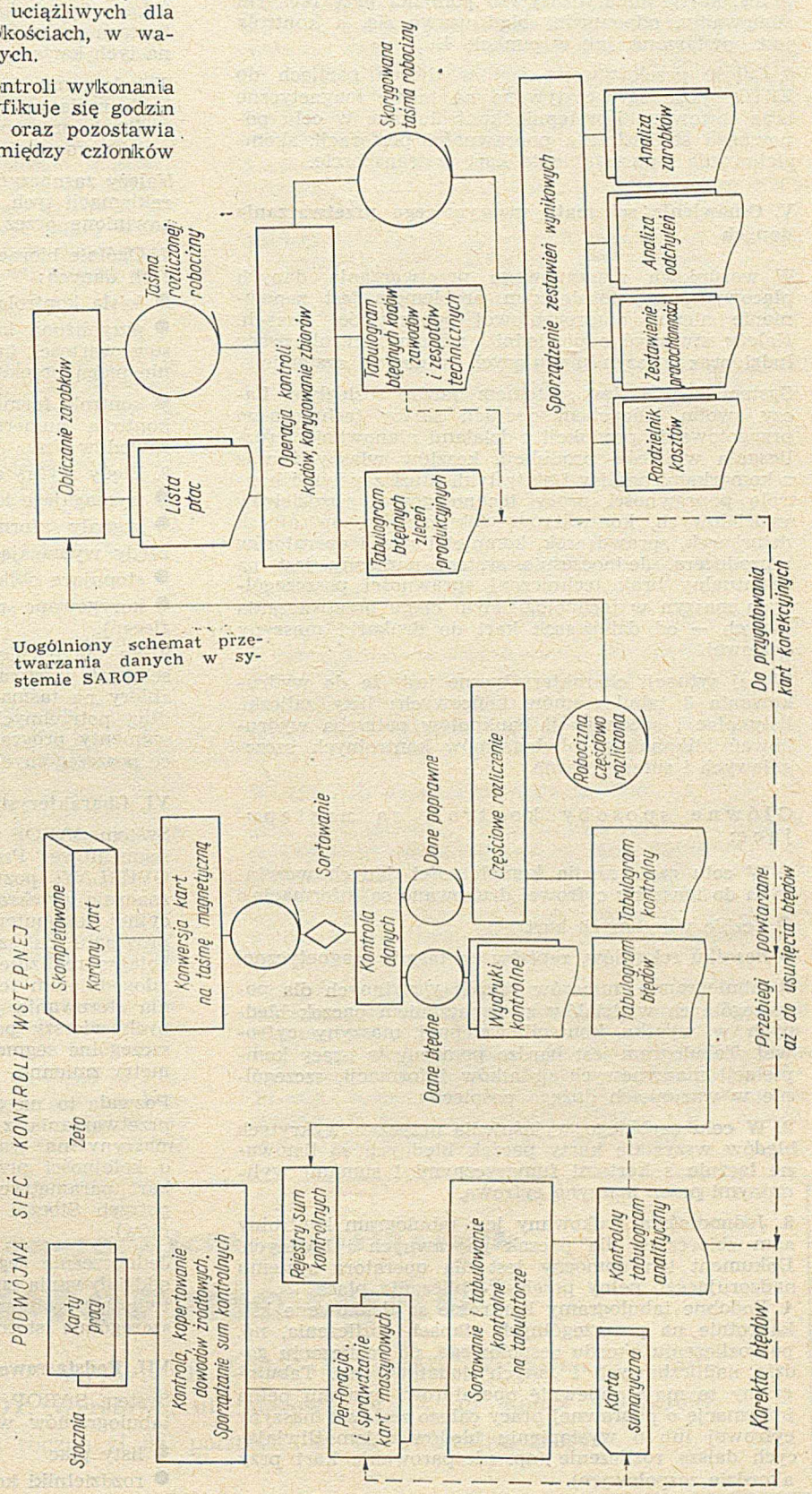
Karty pracy wypisane są na maszynach do pisania przez poszczególne wydziałowe rozdzielnie robót. Po zakończeniu danych operacji i po zarejestrowaniu przekazywane są partiami do stoczniowego Ośrodka Przetwarzania Informacji.

Karty te podlegają następującym czynnościom „obróbki" wstępnej:

1. Przeglądane są w celu stwierdzenia kompletności wszystkich niezbędnych informacji (celowi temu służy specjalny szablon — kontrola formalna).
2. Określone partie kart są kopertowane, co popularnie nazywa się „paczkowaniem" kart. Każda koperta otrzymuje specjalny numer, który jest ważnym nośnikiem informacji kontrolnych oraz istotnym identyfikatorem przy wyszukiwaniu konkretnej karty spośród „stosu" tysięcy kart spływających co miesiąc z poszczególnych wydziałów Stoczni.
3. Każda „paczka" kart podlega kilkakrotnemu specjalnemu sumowaniu celem stworzenia tzw. sum kontrolnych, np. z godzin normowanych, przepracowanych, numerów ewidencyjnych pracowników itp.

Sumy te są niezbędne przy zagwarantowaniu poprawnego przeniesienia danych na maszynowe nośniki informacji (kontrola perforacji kart na sprawdzarkach nie gwarantuje wystarczającej dokładności). Poza tym powstaje problem kontroli poprawności reprodukcji kart, zagubienia kart itp.

Sumy kontrolne paczek kart są podstawą sporządzania sum kontrolnych w układzie poszczególnych wydziałów oraz całej Stoczni. Liczby te będą dalej niezbędne przy kontroli przebiegu obliczeń na komputerze oraz do „uzgadniania" czyli kontroli wynikowej poszczególnych list płac, tabulogramów rozliczeniowych itp. Kontrola ta zostanie szerzej opisana w dalszej części artykułu.



Uogólniony schemat przetwarzania danych w systemie SAROP

4. Po wydziurkowaniu i sprawdzeniu każda paczka kart podlega tabulowaniu na tabulatorze T 320, gdzie sporządzane są poszczególne sumy kontrolne oraz dziurkowane karty sumaryczne, będące później podstawą do kontroli wczytywania kart do maszyny cyfrowej.

5. Każdy tabulogram z tabulatora jest kontrolowany w Sekcji Kontroli z sumami uzyskanymi przy ręcznym sumowaniu. Paczki „zgodne" kompletowane są do wysyłki do ZETO. W przypadku stwierdzenia niezgodności, spowodowanych błędną perforacją karty pracy, karty sumarycznej lub pomyłką przy ręcznym sumowaniu, odpowiedni błąd usuwa się, a kontrola jest powtarzana, jak w punkcie 4.

6. Całość przekazywana jest w dwóch partiach do ZETO, gdzie są wczytywane na taśmy magnetyczne oraz sortowane i wstępnie kontrolowane w celu ponownego stwierdzenia poprawności perforacji, skompletowania i niezagubienia kart w transporcie.

V. Omówienie schematu maszynowego przetwarzania danych

W warunkach maszynowego przetwarzania danych płacowych, najistotniejszym problemem jest zapewnienie niemal stuprocentowej poprawności danych. Należy stworzyć odpowiedni system kontroli pracy ludzi oraz maszyn pracujących w danym systemie.

System SAROP jest systemem bardzo rozległym. Łączy swoim łańcuchem operacji służby technicznego przygotowania produkcji z działami planowania i rozliczania wielkości produkcji, kosztów własnych oraz z komórkami analizy i statystyki. Musi zapewnić kontrolę poprawności pracy technologów i rozdzielców wydziałowych, kontrolerów dokumentów, operatorek dziurkarek, sprawdzarek, kompletatorów i operatorów reproducera. Jednocześnie system narażony jest na ewentualny brak technicznej sprawności poszczególnych maszyn w technologicznym ciągu przetwarzania danych — od dziurkarek kart do drukarki maszyny cyfrowej.

W tej sytuacji charakterystyczne jest, że do wydrukowania 3 tabulogramów końcowych: listy zaliczki, listy płac i specyfikacji banknotów potrzeba wydrukować kilkanaście tabulogramów kontrolnych szczegółowych i sumarycznych.

Główne sposoby kontroli są następujące:

1. W celu zapewnienia kompletności danych wczytanych do maszyny cyfrowej drukowane są informacje o

- liczbie wczytanych kart
- liczniku rekordów, zapisów na taśmie magnetycznej
- tabulogramie numerów paczek wczytanych dla poszczególnych wydziałów z zaznaczeniem paczek błędnych w wyniku kontroli wstępnej maszyny cyfrowej. Tabulogram jest bardzo pomocny w pracy kompletacji maszynowych nośników informacji, szczególnie w warunkach dużego pośpiechu.

2. W celu szybkiego wyjaśnienia przyczyn wykrytych błędów wszystkie karty paczek błędnych są listowane łącznie z kartami sumarycznymi i sumami wyliczonymi przez maszynę cyfrową.

3. Jednocześnie drukowany jest tabulogram kontrolny sum zbiorczych dla paczek poprawnych i błędnych. Dokument ten pomocny jest dla operatora systemu nadzorującego pełny przebieg obliczania płac.

4. Podobne tabulogramy kontrolne są drukowane kilkakrotnie na poszczególnych etapach rozliczenia, np. po rozliczeniu akordu zespołowego, po rozliczeniu godzin nadliczbowych i innych dodatków itp. Tabulogramy te mają zapewnić operatorowi systemu pełną informację o poprawnej pracy całego zestawu maszyny cyfrowej lub o wystąpieniu błędów uniemożliwiających dalsze rozliczenie (np. niesparowanie kart przy akordzie zespołowym).

W przypadku awarii maszyny cyfrowej na dalszym etapie obliczeń pozwalają one również szybko zlokalizować etap wymagający powtórzenia oraz miejsce wznowienia działania programu.

5. Specjalne przeznaczenie mają tzw. liczniki rekordów zbiorów na taśmach magnetycznych, drukowane przez drukarkę wierszową. Dają one informacje o poprawności działania programów sortowania lub łączenia zbiorów na taśmach magnetycznych.

6. Dalszym sposobem kontroli jest drukowanie (na żądanie) specjalnych tabulogramów szczegółowych, informujących każdego pracownika o tym, na jakich kartach pracy figurował i jakie są składniki zarobków na tych kartach.

Tabulogramy te są bardzo przydatne w przypadkach reklamacji ze strony robotników lub w przypadkach przerzutów zarobków spowodowanych błędnymi numerami ewidencyjnymi pracowników.

Należy zaznaczyć, że po kilku miesiącach eksploatacji reklamacji tych jest niezmiernie mało. Rzadko też są zawinione przez „aparat" przetwarzania.

7. Ogólnie biorąc stosowane są 3 główne sposoby kontroli danych:

- ścisła kontrola za pomocą sum kontrolnych paczek
- przybliżona kontrola logiczna lub kontrola zakresów wartości (np. kontrola, że godziny nadliczbowe nie mogą przekraczać godzin przepracowanych)
- kontrola formalna i logiczna kodów (indykacji), np. kontrola numerów zleceń produkcyjnych, numerów zawodów itp.

8. Błędy wykryte programowo dzielą się na:

- wymagające korekty
- sygnały informacyjne.

Błędy wymagające korekty dzielą się z kolei na:

- stopujące obliczenia płac
- korygowane po obliczeniu list płac (np. korekta zleceń).

9. Po „przebrnięciu" przez wszystkie sita kontrolne, schemat przetwarzania jest bardzo prosty: poprawne zbiory na taśmach magnetycznych są sortowane według potrzebnych kryteriów, z których poszczególne segmenty programu powodują wypisanie na drukarce poszczególnych tabulogramów wynikowych.

VI. Charakterystyka programów

System SAROP posiada 3 specjalne programy wielosegmentowe. Programy te są napisane w technice OVERLAY, pozwalającej na pisanie programów znacznie przekraczających pojemność pamięci operacyjnej komputera. Technika ta polega na podziale programu na szereg segmentów, których wykonanie kolejno realizuje obliczenia danego systemu. Dalszym udogodnieniem tej techniki jest możliwość zastosowania sterowania przebiegiem pracy systemem specjalnych wierszy parametrycznych, które wywołują poszczególne segmenty programu i podają pewne parametry zmienne.

Pozwala to na automatyczne sterowanie przebiegiem przetwarzania z minimalną ingerencją operatorów maszyny na wariantowanie obliczeń (każdorazowo o kolejności przetwarzania decyduje ciąg ułożonych kart parametrycznych, dopasowanych do aktualnych potrzeb Stoczni w zakresie sporządzania tabulogramów).

Jednocześnie segmentacja programów ogranicza pole oddziaływania zmian w systemie płac w Stoczni i ułatwia przerabianie programów. Okoliczność ta jest szczególnie istotna przy systemach płacowych.

VII. Podstawowe wzory zestawień wynikowych

System SAROP umożliwia otrzymywanie kilku grup tabulogramów w różnych układach, a mianowicie:

- listy płac
- rozdzielniki kosztów robocizny

- rozliczenia czasu i funduszu płac
- zestawienia pracochłonności produkcji
- analizy zarobków i wydajności pracy robotników
- analizy kosztów odchyleń od normalnego procesu technologicznego.

A oto krótka charakterystyka podstawowych typów tabulogramów:

1. Lista płac zredagowana została w ten sposób, aby maksymalnie wykorzystać pojemność 1 strony tabulogramu (na 1 arkuszu drukowane są 4 wielowarstwowe odcinki płac). Ze względu na brak realnych możliwości uzyskania nadruku nagłówków składników płac na papierze dwuwarstwowym stosuje się drukowanie automatyczne na papierze czystym. Nie jest to rozwiązanie „obojętne" dla drukarki wierszowej oraz czasu drukowania. Ze względu na zbyt małą szerokość wałka drukarki należało stosować szereg skrótów (liczba liter w nagłówkach dopasowana jest do maksymalnej pojemności cyfrowej danego składnika płac) oraz skomasować mniej ważne składniki do rubryk zbiorczych. Odcinek dopasowany jest do rozliczenia płac netto i stanowi gotową podstawę do wypłaty.

2. Rozdzielniki kosztów

Przy projektowaniu tych tabulogramów kierowano się zasadą maksymalnego odciążenia przyszłego użytkownika od pracy ręcznej na tabulogramie przepisywania, przeliczania itp.

Udało się doprowadzić do tego, że poszczególne zlecenia rozliczane są aż do wydziałowego kosztu przerobu (robocizna bezpośrednia + narzuty kosztów wydziałowych), a niektóre rozdzielniki kosztów wliczone są do globalnego kosztu przerobu (z narzutami kosztów ogólnozakładowych).

3. Zestawienia pracochłonności

Odpowiadają poza zasadą wyżej opisaną zasadzie porównywalności informacji otrzymywanych przez różnych użytkowników oraz zasadzie wyeliminowania sprawozdawczości wewnętrznej: poszczególni użytkownicy otrzymują różne kopie tego samego tabulogramu. Dzięki elastycznemu programowaniu, zaprojektowano tabulogramy dopasowane do prowadzenia dodatkowych przeliczeń ręcznych (np. rejestry narastające, uwagi o wyczerpaniu limitu pracochłonności), których automatyzacja okazała się nieopłacalna lub niemożliwa w obecnych warunkach.

VIII. Ogólne uwagi o wdrożeniu systemu

Doświadczenia uzyskane w Stoczni upoważniają do stwierdzenia kilku niezbędnych warunków do osiągnięcia sukcesu:

1. Włączenie przyszłych głównych użytkowników systemu, począwszy od etapu wstępnych założeń do czasu zakończenia wdrażania systemu.

2. Włączenie działów przetwarzania, zarówno ze strony Stoczni jak i ZETO na etapie precyzowania koncepcji organizacji programów oraz przy opracowaniu sposobów kontroli korygowania danych itp. Należy podkreślić, że wdrożenie systemu SAROP autorzy uważają za wspólny sukces dużego grona osób z terenu Stoczni oraz pracowników Działu Przetwarzania ZETO Gdynia.

3. Odpowiednie przeszkolenie komórek współdziałających przy eksploatacji systemu (np. pisarzy warsztatowych kart pracy) w zakresie wymagań i możliwości systemu.

IX. Krótko o kosztach i efektach systemu

Przygotowanie systemu przez 3 — 5 osób (projektantów i programistów) trwało około 2 lat. Wymagało około 250 godzin pracy maszyny cyfrowej na testowanie i uruchomienie systemu. Rozruch trwał około pół roku. Polegał na przetwarzaniu pilotowanym (na części danych z poprzedniego okresu obrachunkowego) oraz równoległym na 1/3 danych w skali Stoczni.

Pozwoliło to na wszechstronne sprawdzenie i uniknięcie niespodzianek przy pierwszym samodzielnym przetwarzaniu.

Jednocześnie był to okres przestawienia się pracowników stoczniowego ZOPI na nowy system.

Był to pierwszy system liczony za pomocą komputera i stanowił zupełne „novum" dla pracowników znających MLA ARITMA.

W związku z tym dla pracowników ZOPI oraz innych komórek Stoczni eksploatacja systemu SAROP ma znaczenie psychologiczne: umożliwia poznanie i nabranie zaufania do systemu automatycznego przetwarzania danych.

Tabulatory odegrały rolę najlepszego agitatora dla spraw stosowania ETO w Stoczni.

Dalszymi efektami są:

1. Zwiększenie ilości, jakości i szybkości informacji analitycznej

2. Polepszenie jakości porównywalności tabulogramów

3. Zmniejszenie pracochłonności dodatkowego przetwarzania informacji z tabulogramów — typowe przy stosowaniu MLA.

Eksploatacja systemu SAROP wymaga około 30 godzin pracy maszyny cyfrowej miesięcznie.

X. Kierunki dalszego rozwoju systemu

Przy przygotowaniu i wdrożeniu systemu SAROP osiągnięto w nakładach pewien „próg systemowy", co oznacza, że przy niewielu dodatkowych nakładach pracy programistów oraz czasu komputera można z tych samych zbiorów danych uzyskać szereg dalszych układów tabulogramów. Stanowi to jeden kierunek rozwoju systemu SAROP.

Dalszym — istotniejszym — kierunkiem doskonalenia będzie zastosowanie doświadczeń autorów i komórek eksploatujących system przy opracowaniu i wdrażaniu branżowego systemu jednostek przetwarzania z dziedziny tematycznej KADRY — ZATRUDNIENIE — PŁACE. System branżowy będzie eksploatowany we wszystkich stoczniach produkcyjnych w oparciu o nowy komputer przemysłu okrętowego.

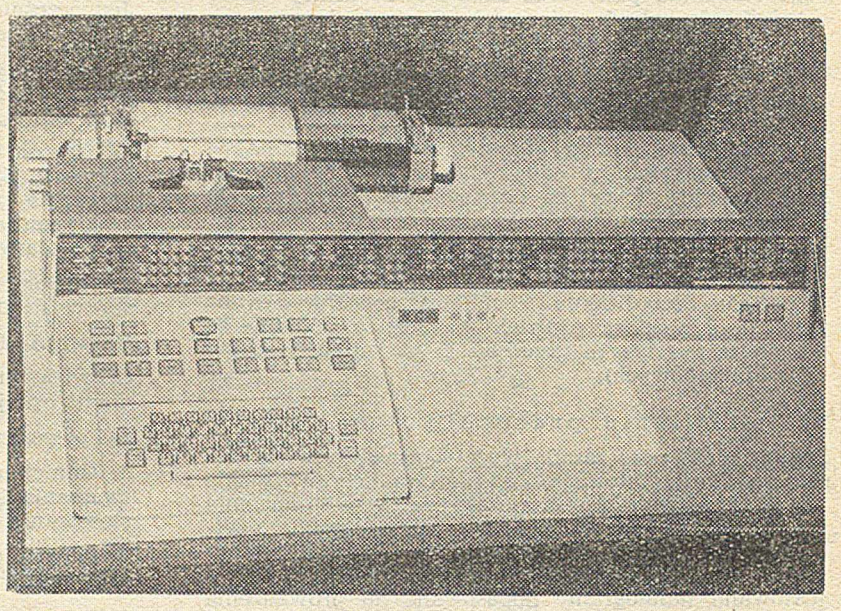
Prenumerujcie miesięcznik

„Informatyka"

Zamówienia na prenumeratę przyjmuje **Zakład Kolportażu Wydawnictw Czasopism Technicznych NOT**, Warszawa, ul. Mazowiecka 12, nr konta 1-9-121697.

Prenumerata normalna roczna zł 96, półroczna zł 48, kwartalna zł 24.

Mgr Zbigniew Ładoś ukończył Akademię Handlową w Krakowie w latach 1945—47. W Narodowym Banku Polskim pracuje od roku 1949. Od roku 1958 działa w NBP na odcinku zastosowań komputerów. W roku 1963 był współorganizatorem Centrum Elektronicznego NBP, w którym obecnie kieruje pracami dotyczącymi programowania komputerów.



ZBIGNIEW ŁADOŚ
Centrum Elektroniczne
Narodowego Banku Polskiego
Warszawa

Uruchamianie nowych programów na komputerze NCR 315

681.322.06

Z praktyki Centrum Elektronicznego Narodowego Banku Polskiego w Warszawie

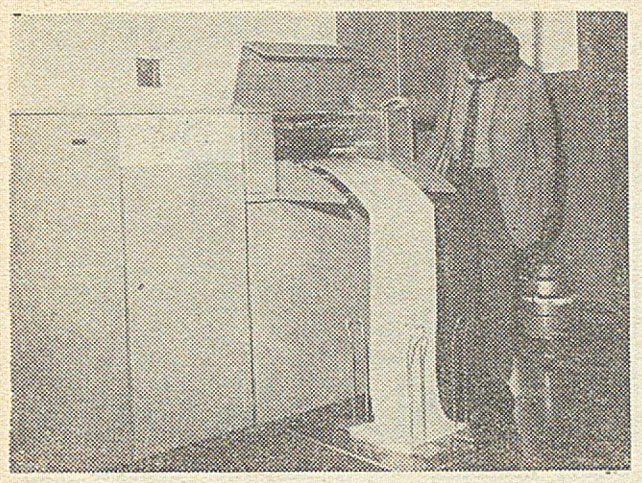
Przedstawiono warunki organizacyjne niezbędne do właściwego pisania i uruchamiania programów. Omówiono system operacyjny NACT do automatycznego uruchamiania programów na komputerze NCR 315.

Problem uruchamiania programów

Uruchamianie nowych programów, które mają służyć do częstego przetwarzania danych masowych, jest wprawdzie pracą trudną, ale zarazem bardzo interesującą, a niekiedy — w przypadkach bardziej skomplikowanych programów — wręcz pasjonującą.

W niniejszym artykule przedstawiono niektóre charakterystyczne okoliczności posiadające istotny wpływ na przebieg uruchamiania nowych programów do przetwarzania danych na maszynie NCR 315 w Centrum Elektronicznym NBP w Warszawie.

Wobec bardzo wielu zadań wykonywanych użytkowo na tej maszynie (od 5 lat intensywna eksploatacja na 3 zmiany) konieczne jest szczególnie oszczędne gospodarowanie czasem jej pracy. Na każdy program wyznacza się programiście limit czasu potrzebny na kompilacje i testowanie programu. Często czas maszyny na uruchamianie nowych programów jest do dyspozycji jedynie w nocy lub w innym nie plano-



wanym czasie (np. gdy powstało opóźnienie w dostarczaniu danych do przetwarzania użytkowego).

Dlatego też stosuje się z reguły zasadę uruchamiania nowych programów zaocznie, tzn. bez udziału programisty. Programista jest zainteresowany w takim przygotowaniu przebiegu uruchamiania programów, aby jak najmniej użyto czasu komputera, szczególnie wówczas, gdy program jest bardzo skomplikowany i będzie wymagał częstego testowania (sprawdzania) na specjalnie przygotowanych danych przykładowych, zapewniających symulację użytkowego przetwarzania.

Niekiedy wysuwa się zarzuty, że programiści popełniają zbyt dużo błędów przy pisaniu programów, co jest przyczyną przedłużania się czasu uruchamiania programu i powoduje większe koszty zużycia czasu maszyny. Trudno jest ustalić w tym zakresie ścisłe normy, jednakże można w przybliżeniu określić zarówno czas uruchamiania programu, jak i przewidywane zużycie czasu komputera. Odchylenia od wyznaczonych limitów wynikają z tego, że nigdy nie można dokładnie przewidzieć, ile i jakie błędy popełni programista w trakcie opracowywania programu. Nie zdarza się w praktyce, aby do napisanych programów nie zakradł się jakiś błąd. Określenie „błąd" należy w tym przypadku rozumieć bardzo szeroko, a więc są to nie tylko zwykłe błędy, związane z użyciem języka programowania, ale również pewne przeoczenia w założeniach do programu, które niekiedy ujawniają się dopiero przy testowaniu. Dokładne sprawdzenie całego programu jest zawsze konieczne bez względu na jakość pracy programisty. Wyniki testowań programu muszą być opisane i dołączone do dokumentacji programu. Czasokres uruchamiania programu i zużycie czasu komputera zależy m.in. od:

a) zakresu i rodzaju zadań realizowanych przez program

b) ilości powiązań logicznych w programie, ilości informacji stałych (opisów), które trzeba wprowadzić do programu, stopnia komplikacji programu itp.

c) możliwości częstego testowania programu

d) kwalifikacji zawodowych programującego.

Zmniejszenie zużycia czasu maszyny w każdej pracy zależy m. in. od minimalizacji interwencji operatora obsługującego komputer. Najszybszy w pracy operator jest zawsze zbyt powolny, biorąc pod uwagę koszt straconego czasu maszyny na niezbędne interwencje. Jeszcze kosztowniejsze są pomyłki operatora przy obsłudze maszyny. Dlatego przy uruchamianiu nowych programów należy w miarę możliwości rezygnować z pracochłonnej obsługi operatora i dążyć do automatyzacji tych czynności.

Tak zwane „krokowe" sprawdzanie z pulpitu sterującego maszyny kolejnych rozkazów testowanego programu z reguły nie jest już dawno stosowane.

Programy uruchamiane w Centrum Elektronicznym NBP przeważnie służą do terminowego przetwarzania masowych danych bez udziału programisty. Z tych względów programy muszą być wszechstronnie opracowane i przetestowane, aby w czasie przetwarzania nie było nieoczekiwanej przerwy, konieczności wzywania programisty, szukania błędu itd. Spowodowałoby to niedotrzymanie terminu przetwarzania i różne poważne konsekwencje organizacyjne u użytkowników. Przy masowych operacjach księgowych, statystycznych itp. — mimo dominującej powtarzalności typowych przypadków — istnieje zawsze pewne prawdopodobieństwo pojawienia się wyjątków, tj. przypadków nietypowych, np. bardzo trudnych do przewidzenia rodzajów błędów w danych wejściowych.

Program musi więc być bardzo wszechstronnie przetestowany, aby nie było w nim żadnych nieprzewidzianych sytuacji.

Na sprawny przebieg uruchamiania programów w pierwszym rzędzie ma wpływ autor programu, a następnie odpowiedni system operacyjny, dostarczany przez producenta maszyny. W tym względzie producenci prześcigają się w opracowywaniu różnych, rzekomo rewelacyjnych programów, ułatwiających programiście jego pracę.

Wykazy oprogramowania dostarczanego przez producenta są niekiedy bardzo obszerne, lecz w praktyce tylko niewielka część tych programów jest częściej używana. Dotychczas nie odkryto jeszcze żadnej skutecznej metody, która by radykalnie zmniejszała wysiłek, jaki musi włożyć programista w tej fazie pracy nad programem.

Maszyna NCR 315 posiada specjalny system operacyjny, służący głównie do uruchamiania programów, zwany NACT (skrót od THE NATIONAL AUTOMATIC CONTROLER FOR TESTING CRAM PROGRAMS NEAT ONE).

Automatyczne uruchamianie programów

Jak już wspomniano w Centrum Elektronicznym Narodowego Banku Polskiego obok cyklicznego przetwarzania masowych danych powodującego pełne obciążenie maszyny na trzy zmiany, są kontynuowane prace nad uruchamianiem nowych programów, dla dalszych zadań wprowadzanych do eksploatacji na maszynie NCR 315. Jest to możliwe dzięki usprawnianiu i optymalizowaniu dotychczas eksploatowanych programów. Ponadto do wielu programów użytkowych trzeba na zlecenie użytkowników okresowo wprowadzać różne zmiany. W miarę wzrostu wymagań użytkowników, programy stają się coraz trudniejsze do opracowania, tym bardziej, że równocześnie — dla zminimalizowania czasu przetwarzania muszą one być bardzo optymalnie rozwiązane. Dla prac przetwarzanych użytkowo w cyklu codziennym — programy muszą być szczególnie optymalne i niezawodne w działaniu. Wszystkie te okoliczności uzasadniają konieczność odpowiedniego zorganizowania uruchamiania nowych programów.

Kolejne etapy tych prac ilustruje tablica.

W dalszym ciągu zostanie opisany tylko fragment prac, związany z testowaniem programu (punkt 17 tablicy) za pomocą specjalnego programu nadzorczego NACT zsynchronizowanego z całą serią programów operatorskich i innych programów standardowych, znajdujących się we wspólnej bibliotece na kartach magnetycznych. Jest to więc specjalny system operacyjny dla automatycznego uruchamiania programów. Działanie NACT jest sterowane za pomocą kart perforowanych. W przerwach między wczytywaną jedną lub kilkoma kartami sterującymi są wykonywane różne programy. Przebieg testowania ilustruje następujący przykład:

Obok sali komputera na stoliku leżą teczki. W każdej teczce jest niewielki plik kart perforowanych i „zlecenie pracy" na maszynę, w którym programista podaje m.in. numery potrzebnych szpul taśm magnetycznych, zasobników kart magnetycznych oraz planowany czas na uruchomienie programu.

Komputer realizuje w zasadzie bez udziału operatora następujące funkcje:

a) wczytywanie danych testowych (na kartach) i automatyczne ich wprowadzanie na taśmę lub na karty magnetyczne

b) pobieranie testowanego programu z kart magnetycznych do pamięci operacyjnej z tym, że uprzednio może być automatycznie wyzerowana pamięć, po wczytaniu mogą być do programu automatycznie wniesione poprawki, zawarte w kartach perforowanych

c) testowanie programu, ewentualnie z odpowiednio sterowanym śledzeniem i z drukowaniem przez drukarkę adresów wszystkich lub tylko niektórych rozkazów

d) drukowanie w całości lub fragmentami zawartości nowo utworzonych zbiorów danych z taśm lub z kart magnetycznych

e) zmienianie zawartości fragmentów zbiorów danych w kartach magnetycznych celem przetestowania innych, dotychczas jeszcze nie przebadanych fragmentów kontrolowanego programu (zmiany dotyczące danych są zawarte w kartach perforowanych)

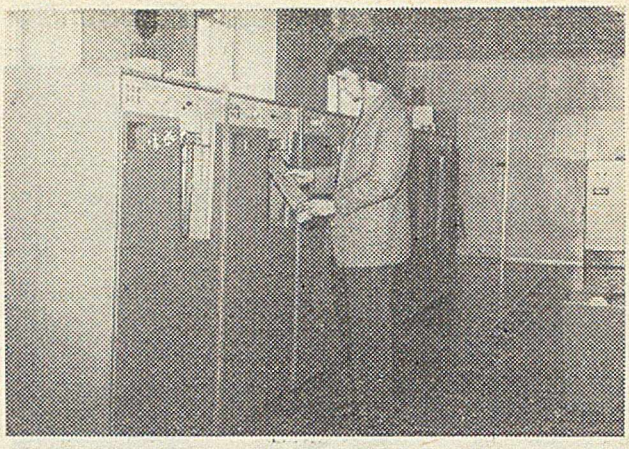
f) powtarzanie fragmentów od punktu b) do punktu d) w celu prześledzenia działania programu w zmienionych warunkach

g) powtarzanie fragmentów od b) do f) zależnie od potrzeb, przy czym każdorazowo są testowane inne

Tablica

Czynności związane z opracowaniem i uruchomieniem programu przetwarzania danych

Lp.	Czynności	Gdy błąd powrót do lp.
1	Analiza założeń programu	
2	Opracowanie ogólnego schematu blokowego, wskazującego na organizację programu	
3	Weryfikacja ad 2	2
4	Opracowanie szczegółowe schematu blokowego programu, wskazującego na wszystkie powiązania logiczne w programie. Podjęcie opracowania dokumentacji programu	
5	Weryfikacja ad 4	4
6	Opracowanie programu w języku zewnętrznym	
7	Weryfikacja formalna programu ad 6	6
8	Perforowanie kart dla programu ad 6	
9	Opracowanie specjalnych danych do testowania programu	
10	Weryfikacja danych do testowania programu	9
11	Perforowanie kart lub taśm z danymi testowymi	
12	Opracowanie programu rozkazów sterujących przebiegiem testowania (NACT)	
13	Weryfikacja ad 12	12
14	Perforowanie kart ad 12	
15	Formalne sprawdzenie prawidłowości wyperforowania kart ad 14, 11, 8	14, 11, 8
16	Kompilacja programu i usuwanie formalnych błędów	16
17	Uruchamianie programu na danych testowych, wyszukiwanie błędów, korygowanie programu i dokumentacji (o ile nie ma innych możliwości — ręczne wyliczanie wyników pracy programu i porównanie z wynikami pracy maszyny)	17,16
18	Uruchamianie programu przy współdziałaniu z innymi programami na danych testowych	17, 16
19	Zakończenie opracowania dokumentacji programowej i eksploatacyjnej	
20	Dublowanie na maszynie prac wykonywanych metodą tradycyjną	19, 17, 16
21	Weryfikacja dokumentacji	20, 19
22	Kontrola i odbiór przez użytkownika	
23	Usuwanie błędów wykrytych przy odbiorze programu	17, 16
24	Wprowadzanie ewentualnych zmian do programu na dodatkowe zlecenie użytkownika	1



fragmenty badanego programu; program może być w każdym przebiegu dodatkowo modyfikowany w różny sposób.

Istnieją oczywiście jeszcze inne możliwości, np. automatyczne kopiowanie zbiorów z kart na taśmy magnetyczne, z jednych kart na inne karty magnetyczne itp. Ponadto można kasować zawartość kart magnetycznych, drukować tylko metryczki zbiorów itd. Szacuje się, że gdyby wszystkie te czynności operator miał sterować ręcznie, wtedy ogólne zużycie czasu maszyny byłoby kilkakrotnie większe.

W miarę potrzeby, dodatkowe standardowe programy pomocnicze są we własnym zakresie opracowywane w Centrum Elektronicznym NBP dla różnych użytkowników maszyny NCR 315. W każdym takim programie przewiduje się możliwość podawania różnorodnych parametrów sterujących, zarówno z kart perforowanych, jak również (na życzenie operatora) z maszyny do pisania w pulpicie sterującym komputera. Jest to szczególnie dogodne, np. w przypadku uszkodzenia karty perforowanej, gdyż operator może wówczas „ręcznie" sterować działaniem tych programów.

Stosowanie systemu automatycznego testowania programów NACT wymaga używania specjalnych bibliotek programów zapisanych na kartach magnetycznych łącznie z zestawem programów współpracujących z NACT oraz z programami podlegającymi testowaniu. Biblioteki programów na kartach magnetycznych lepiej nadają się do tego celu od bibliotek na taśmach magnetycznych, albowiem nie trzeba przewijać szpuli taśmy do poszukiwania co chwilę innego programu. Z kart magnetycznych programy są wywoływane metodą bezpośredniego dostępu.

Po kompilacji programu opracowanego przez programistę można za pomocą NACT automatycznie wprowadzić przetłumaczony program do biblioteki programów i od razu testować w jednym przebiegu bez interwencji operatora maszyny.

NACT umożliwia też wprowadzenie przy tej okazji różnych zmian do programu znajdującego się w bibliotece. W wyniku działania systemu NACT programista może otrzymać — zależnie od potrzeb: drukowane wyniki działania całego lub części programu, drukowaną zawartość fragmentu lub całej pamięci operacyjnej lub pamięci zewnętrznych i różnych rejestrów, drukowane spisy metryczek zbiorów danych, programów itp.

Jest to wystarczająca dokumentacja dla pełnej oceny działania testowanego programu. Równocześnie wymaga się od programisty, aby nie powodował drukowania przez komputer nadmiernych ilości zestawień kontrolnych.

W odpowiedzi na ewentualne zarzuty, że NACT wymaga każdorazowo specjalnych kart sterujących —

trzeba wyjaśnić, że przeważnie wykorzystuje się przy każdym testowaniu większość tych samych kart, zmieniając tylko niektóre lub układając je w innej kolejności.

NACT wykorzystuje się też do sprawdzania większej liczby programów na wspólnych danych testowych. Poprawne działanie NACT jest uzależnione od bezbłędnego przygotowania przez programistę kart sterujących, które z reguły powinny być sprawdzone. NACT jest również dogodny do testowania programu wykonywanego w obecności programisty, albowiem umożliwia odpowiednie przygotowanie testowania. W wyjątkowych przypadkach, np. jednorazowego testowania jakiegoś programu po wprowadzeniu do niego modyfikacji wynikających ze zmiany przepisów, nieopłacalne jest korzystanie z systemu NACT i wówczas z tej samej biblioteki programów można indywidualnie wywoływać tylko dany program.

Inne problemy związane z uruchamianiem programów

Wielorakie czynności programisty w okresie uruchamiania programu wymagają dobrej organizacji pracy. Jest to bowiem zasadniczy warunek wykonania tych prac w planowym terminie. Ponadto programista powinien stale dążyć do opracowania bardzo dokładnej, przejrzystej i uporządkowanej dokumentacji programu.

W tym zakresie obowiązują w Centrum Elektronicznym NBP specjalne przepisy.

Dokumentację tę sporządza programista od początku prac nad programem, a kończy w ostatnim dniu testowania programu. Do dokumentacji są włączone tabulogramy z drukarki, na podstawie których można udowodnić, w jakim zakresie program został sprawdzony.

Poważnym niebezpieczeństwem dla jakości programu już sprawdzonego na maszynie są zmiany wprowadzone na życzenie użytkownika. Przy zmianie jakiegoś fragmentu programu można przeoczyć konieczność zmiany innego fragmentu lub też zmieniając różne fragmenty bardzo dużego programu — uszkodzić pewne jego funkcje, niezbędne przy przetwarzaniu innego typu danych. W związku z tym po takich poprawkach program powinien zostać dokładnie przetestowany na maszynie.

W związku z powyższymi uwagami wydaje się wskazane, aby przy odbiorze opracowanych lub modyfikowanych programów, użytkownik każdorazowo sprawdzał otrzymany do wykorzystania system elektronicznego przetwarzania danych. Do tego sprawdzania powinien użytkownik przygotować specjalne dane modelowe, uwzględniając wszystkie przypadki, jakie mogą wystąpić w praktyce. Również w okresie tzw. dublowania na komputerze prac wykonywanych metodą tradycyjną, ujawnione zostaje często szereg przeoczeń w programach, które dotyczą przeważnie postępowania w przypadkach wystąpienia w danych nietypowych błędów.

Okazuje się w praktyce, że najbardziej wnikliwi projektanci i programiści nie są w stanie przewidzieć wszystkich wariantów błędów, jakie popełniają osoby wypełniające dokumenty źródłowe stanowiące podstawę do przygotowania maszynowych nośników informacji.

Wyłonił się więc dyskusyjny problem, w jakim zakresie ośrodek obliczeniowy powinien być odpowiedzialny za bezbłędność opracowanych programów do przetwarzania danych.

Szczupłe ramy tego artykułu nie pozwalają na szersze opisanie tego tematu, który ma różne aspekty, m.in. również aspekty formalno-prawne. Nasuwa się tu analogia do sprzedaży bardziej skomplikowanych urządzeń, które są wprawdzie wszechstronnie sprawdzane przez producenta, ale dalszy ciąg sprawdzania następuje u odbiorcy w okresie gwarancyjnym. Interesująca jest w tym względzie praktyka zagranicznych przedsiębiorstw specjalizujących się wyłącznie w opracowywaniu i sprzedaży programów do przetwarzania danych. Przedsiębiorstwa te udzielają około półrocznej gwarancji, w czasie której bezpłatnie są usuwane błędy wykryte przez użytkownika. Wydaje się, że praktyka mobilizuje użytkownika do bardzo wszechstronnego sprawdzania zakupionych programów przy odbiorze i w okresie gwarancyjnym.

Na zakończenie trzeba wyrazić życzenie, aby szczególnie trudna i odpowiedzialna praca programisty, specjalizującego się w skomplikowanych problemach przetwarzania danych, spotkała się z pełnym zrozumieniem u użytkowników systemów EPD. Użytkownicy ci często dziwią się, że opracowanie programów jest zadaniem tak czasochłonnym i że zgłaszanie jakichkolwiek zmian założeń do programów wymaga każdorazowo żmudnych przeróbek w programach i ponownego testowania programów. Utarło się ostatnio określenie „konserwacja" programów, oznaczające dostosowywanie programów do zmieniających się życzeń użytkowników. Żmudne prace przy testowaniu uzasadniają konieczność zwrócenia większej uwagi na bardziej perspektywiczne i wszechstronne ujmowanie potrzeb użytkowników jeszcze przed rozpoczęciem programowania, co powinno zapewnić większą stabilność programów. Wydaje się, że przy niedostatecznie stabilnych założeniach dla skomplikowanych programów nie powinno się podejmować programowania, nawet w przypadku pewnego zaangażowania środków we wstępne prace projektowe.

Z Życia Stowarzyszenia Elektryków Polskich

W marcu 1971 roku odbyło się w Wielkopolskich Zakładach Teletechnicznych TELETRA wspólne posiedzenie prezydiów Komisji Elektronicznego Przetwarzania Informacji SEP i Polskiego Komitetu Automatycznego Przetwarzania Informacji NOT.

Organizatorem tego posiedzenia było koło SEP przy Zakładach TELETRA. Uczestnicy zebrania mieli możność zapoznania się z produkowanymi lub będącymi w stadium wdrażania do produkcji urządzeniami na specjalnie z okazji zebrania przygotowanej wystawie.

W zebraniu udział wzięli również: zastępca przewodniczącego Komitetu Nauki i Techniki prof. dr inż. Zbigniew Jasicki oraz dyrektorzy Krajowego Biura Informatyki — dr Zbigniew Gackowski i dr inż. Andrzej Targowski.

Przedmiotem obrad — poza zagadnieniami działalności i współpracy obydwu jednostek, tj. PKAPI NOT i KEPI SEP — były: rola informatyki w kraju i główne kierunki działalności Krajowego Biura Informatyki.

W wyniku dyskusji wyrażono zgodny pogląd co do konieczności podniesienia rangi informatyki w kraju i rangi związanego z tą dziedziną zawodu. Zdaniem uczestników zebrania realizacja tego rodzaju zadania wymaga rozszerzenia, a zarazem integracji środowiska informatyki. Poważną rolę w tym zakresie powinny odegrać koła zakładowe SEP, m. in. poprzez stwarzanie platformy dyskusji społecznych i możliwie pełne wykorzystywanie form działalności w zakresie podnoszenia kwalifikacji zawodowych w tym środowisku.

Henryka Piasecka

Mgr Antoni Nadolski ukończył w roku 1958 studia na Wydziale Przemysłu Szkoły Głównej Planowania i Statystyki w Warszawie. Bezpośrednio po studiach pracował w administracji w Prezydium Rady Narodowej m.st. Warszawy, a w roku 1964 ukończył Podyplomowe Studium Mechanizacji i Automatyzacji przy SGPiS.
Od roku 1964 jest zatrudniony w Narodowym Banku Polskim — w Centrum Elektronicznym, początkowo jako programista, a obecnie — jako naczelnik Wydziału Przetwarzania Danych.

ANTONI NADOLSKI
Centrum Elektroniczne
Narodowego Banku Polskiego
Warszawa

681.322.004.14:681.322.06.004.14

System eksploatacji programów na komputerze NCR 315

Z doświadczeń Centrum Elektronicznego Narodowego Banku Polskiego w Warszawie

Przedstawiono podstawowe wyposażenie ośrodka obliczeniowego resortu finansów i rodzaje wykonywanych prac. Podano właściwości systemu operacyjnego maszyny NCR 315 oraz sposób jego wykorzystania. Przetwarzanie danych odbywa się z zastosowaniem jednostek pamięci zewnętrznej na kartach magnetycznych.

Organizacja prac realizowanych na elektronicznej maszynie cyfrowej i system eksploatacji programów w warunkach masowego przetwarzania danych są uzależnione od wielu czynników. Szczególne znaczenie mają następujące czynniki:

- konfiguracja zestawu maszyny cyfrowej oraz stopień jej wykorzystania
- system operacyjny tej maszyny
- częstotliwość realizacji na maszynie tych samych tematów, tzw. częstotliwość cyklu przetwarzania
- umiejętność wykorzystania właściwości systemu operacyjnego maszyny
- stosowanie metod programowania ułatwiających i usprawniających obsługę operatorską maszyny.

Na podstawie kilkuletnich doświadczeń eksploatacyjnych Centrum Elektronicznego NBP w Warszawie można scharakteryzować warunki i zasady prawidłowej obsługi prac, realizowanych na maszynie cyfrowej, a zwłaszcza wykorzystanie systemu operacyjnego maszyny NCR 315.

Zestaw maszyny oraz jej wykorzystanie

W Centrum Elektronicznym NBP jest zainstalowana maszyna cyfrowa II generacji NCR 315. Maszyna ta, zakupiona w roku 1964 do potrzeb różnych instytucji resortu finansów składa się z następujących modułów:

- jednostki centralnej wyposażonej w ferrytową pamięć operacyjną o opojemności 20 000 sylab 12-bitowych. Zapis liczb jest dziesiętny, kodowany dwójkowo, tzn. w jednej sylabie można maksymalnie zapisać liczbę 999. Pojemność akumulatora do działań arytmetycznych wynosi 8 sylab. Szybkość działania jednostki centralnej ocenia się na około 20 000 podstawowych rozkazów na sekundę
- czterech jednostek pamięci na kartach magnetycznych typu CRAM/CARD RANDOM ACCESS MEMORY.

Jednostki te wyposażone w wymienne zasobniki z kartami magnetycznymi są wykorzystywane jako podstawowa pamięć zewnętrzna. Szybkość wymiany informacji z pamięcią operacyjną wynosi około 100 tysięcy znaków na sekundę. W jednym zasobniku mieści się 256 kart magnetycznych, na których można zapisać około 8 mln 300 tys. cyfr. Istnieje możliwość bezpośredniego zapisu lub odczytu informacji, zapisanych na jednej z siedmiu ścieżek każdej karty magnetycznej;

- dwóch jednostek pamięci na taśmach magnetycznych. Jednostki te charakteryzują się zapisem 7-ścieżkowym i szybkością przesyłania informacji rzędu 60 tys. znaków/sekundę. Służą do przetwarzania w niektórych pracach oraz do kopiowania z kart magnetycznych długoterminowych zbiorów danych lub do awaryjnego zabezpieczenia zbiorów danych i programów;
- jednej drukarki wierszowej, charakteryzującej się szybkością drukowania rzędu 600—900 wierszy/minutę.

Wykorzystywana jest ta drukarka do drukowania masowych informacji wynikowych;

- jednego czytnika kart dziurkowanych. Czytnik ten charakteryzuje się szybkością odczytu rzędu 400 kart/minutę.

Służy on do wprowadzania niewielkiej ilości danych oraz jako urządzenie wejściowe przy kompilowaniu programów;

- dwóch czytników taśm dziurkowanych. Czytniki te charakteryzują się szybkością odczytu taśmy rzędu 1000 znaków/sekundę. Służą one jako podstawowe urządzenia do wprowadzania masowych danych wejściowych;
- dwóch dziurkarek taśmy papierowej. Dziurkarki te charakteryzują się szybkością działania rzędu 110 znaków/sekundę. Służą one do reprodukcji programów operatorskich przechowywanych na taśmach dziurkowanych.

Maszyna NCR 315 została zainstalowana w lipcu 1965 roku, a od r. 1966 pracuje na trzy zmiany, przy czym czas jej włączenia do sieci elektrycznej wynosi 20 godzin na dobę.

W pierwszym okresie eksploatacji większość czasu była przeznaczona na uruchamianie nowych programów. Od grudnia 1965 roku rozpoczęto przetwarzanie użytkowe, zaś od roku 1967 — większość czasu przeznacza się na przetwarzanie użytkowe prac, głównie z zakresu bankowości i ubezpieczeń.

Centrum Elektroniczne jako ośrodek resortowy obsługuje aktualnie 6 różnych instytucji resortu finansów w następującym zakresie:

Bank Handlowy S. A. w Warszawie — 4 prace dzienne tworzące ponadto cykle przetwarzania dekadowego i miesięcznego

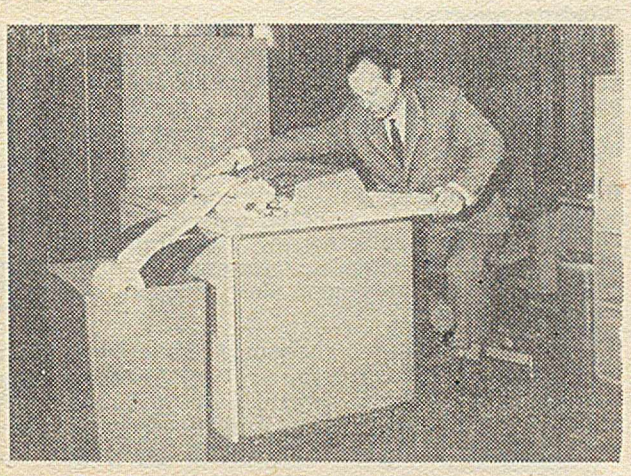
Bank Rolny — 1 praca okresowa

Narodowy Bank Polski — 8 prac, z tego 3 dzienne, tworzące cykle przetwarzania dekadowego, miesięcznego i kwartalnego. Pozostałe 5 są pracami typu okresowego

Ministerstwo Finansów — 2 prace okresowe
Powszechna Kasa Oszczędności — 1 praca wykonywana co drugi dzień tworząca cykle przetwarzania dekadowego, miesięcznego i rocznego.

Państwowy Zakład Ubezpieczeń — 2 prace roczne, wykonywane sukcesywnie w ciągu całego roku z nasileniem pod koniec roku.

W dalszym ciągu termin „praca" będzie używany dla określenia zestawu programów służących do opracowania określonego tematu, np. księgowości i statystyki oddziałów miejskich NBP, ubezpieczeń rolnych PZU, planowania kasowego NBP itp.



Pojęcie „cykl" obejmuje programy stanowiące wycinek pracy, np. cykl dzienny, cykl dekadowy, cykl wejścia, cykl drukowania, cykl sprawozdań miesięcznych itp.

Wyliczone powyżej prace mają różny zasięg i różną budowę, dlatego też trudno byłoby je opisywać szczegółowo w ramach jednego artykułu. Orientacyjnie można podać, że pierwsza praca zorganizowana i przetwarzana na maszynie NCR 315, a obejmująca prowadzenie rachunków bieżących jednego oddziału miejskiego NBP wymagała 3,5 godziny czasu maszynowego. Obecnie ta sama praca, w której podstawowe programy zostały te same, a jedynie w lepszy sposób wykorzystano możliwości systemu operacyjnego maszyny, służy do obsługi 6 oddziałów miejskich NBP w Warszawie, obejmujących około 10 000 rachunków i wymaga dziennie tylko 4—4,5 godziny czasu maszyny.

Praca dla potrzeb PKO, obejmująca około 1200 tys. rachunków oraz 20—30 tys. dziennych operacji wymaga co drugi dzień około 7 godzin czasu maszyny.

Dla wszystkich tych prac opracowano około 1200 programów, które są zgrupowane w 25 bibliotekach programów zarejestrowanych w 25 zasobnikach kart magnetycznych.

Struktura czasu wykorzystania maszyny NCR 315 w roku 1969 przedstawiała się następująco:

• czas włączenia maszyny	6323 godz.	(100%)
• użytkowe przetwarzanie	4511 godz.	(71,2%)
• uruchamianie programów	983 godz.	(15,8%)
• konserwacja maszyny	615 godz.	(9,7%)
• konserwacja materiałów magnetycznych, poprawianie wadliwych usług, awarie, straty na skutek błędów wejścia itp.	214 godz.	(3,3%)

Zużycie 87% czasu włącznia maszyny na użytkowe przetwarzanie i uruchamianie nowych programów lub zmiany w dotychczasowych programach (tzw. konserwacja programów, np. w wyniku zmian przepisów itp.) dobrze świadczy o sprawności jej systemu operacyjnego i dobrej organizacji pracy, biorąc pod uwagę dużą liczbę (1200) i różnorodność eksploatowanych programów.

System operacyjny

W resorcie finansów większość wykonywanych prac składa się z dużej ilości programów łączonych w cykle, które z kolei składają się na całość danej pracy. Wykonanie poszczególnych cykli może być uzależnione od szeregu warunków, jak np. od dnia przetwarzania, jeżeli jest to cykl miesięczny, kwartalny itp., od wyników poprzedniego cyklu i wykonania odpowiednich prac kontrolnych, a co za tym idzie i korekt do utworzonych już zbiorów. Dlatego też przy organizowaniu określonych prac zwrócono szczególną uwagę na wszechstronne wykorzystanie właściwości systemu operacyjnego maszyny NCR 315.

W skład systemu operacyjnego maszyny NCR 315 wchodzą następujące zasadnicze programy standardowe:

LIBRARIAN — program służący do tworzenia bibliotek programów na kartach magnetycznych

SUPERVISOR — program zapewniający możliwość automatycznego wczytywania programów znajdujących się w bibliotekach na kartach magnetycznych oraz umożliwiający wybieranie programów zależnie od lat kalendarzowych i różnych innych warunków, określonych przez programistę

KERNEL — program zajmujący się współpracą jednostki centralnej z jednostkami pamięci CRAM, zapewniający właściwe zwalnianie, odczytywanie i zapisywanie kart magnetycznych oraz sygnalizowanie błędów powstałych przy zapisie lub odczycie tych kart.

ESTEP — odpowiednik programu KERNEL dla taśm magnetycznych

EXTREMITY — umożliwia automatyczne tworzenie lub odczytywanie istniejących zbiorów danych na kartach magnetycznych oraz kontrolę ważności zbiorów poprzez sprawdzenie odpowiednich zapisów w metryczce zbioru. Metryczka taka jest tworzona w momencie powstawania (zakładania) zbioru, a przy każdym odczycie lub likwidacji zbioru — sprawdzana

RESCUE — umożliwia wykonywanie co pewien czas, określony w programie użytkowym, awaryjnych wypisów zawartości pamięci operacyjnej na karty magnetyczne. W razie awarii maszyny umożliwia to rozpoczęcie pracy od ostatniego wypisu pamięci

RESTART — umożliwia uruchomienie programu użytkowego od ostatniego wypisu pamięci, wykonanego przez program RESCUE.

Wykorzystanie systemu operacyjnego przy pracach użytkowych

W organizowanych w Centrum Elektronicznym NBP pracach wszystkie podstawowe operacje związane z przetwarzaniem danych odbywają się przy użyciu jednostek pamięci na kartach magnetycznych. Zasobnik z kartami magnetycznymi służący jako biblioteka może zawierać również zbiory danych wykorzystywane do przetwarzania. Umożliwia to oszczędne użytkowanie zasobników z kartami magnetycznymi. Aby system operacyjny mógł sprawnie działać oraz aby był zawsze w stanie gotowości eksploatacyjnej, każdy zestaw kart magnetycznych stanowiący zawartość zasobnika musi być odpowiednio przygotowany do tego celu.

Wprowadzenie systemu operacyjnego przebiega w następujący sposób. Zapisanie na każdej ścieżce metryczki zawierającej numer zasobnika, numer karty i numer ścieżki.

Wpisanie programów systemowych, tj. KERNEL, EXTREMITY, RESCUE (rys. 1).

Po wykonaniu tych czynności, zasobnik może być przekazany do eksploatacji i mogą być na nim zapi-

Rys. 1. Wprowadzenie systemu operacyjnego niezbędnego przy opracowywaniu zbiorów

sywane zbiory danych. Jeżeli ma stanowić również bibliotekę programów, należy wykonać następujące operacje:

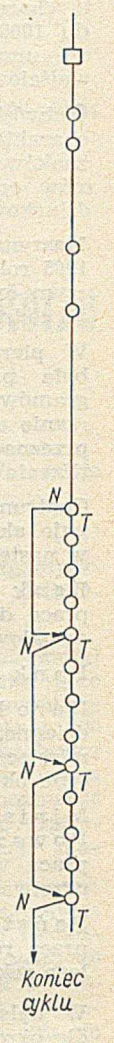
- Zapisać program SUPERVISOR
- Zapisać program LIBRARIAN
- Zapisać programy użytkowe, np. A, B, C, D poprzez przeniesienie z innego zasobnika CRAM, zawierającego programy po kompilacji (rys. 2).

Rys. 2. Wprowadzenie systemu operacyjnego niezbędnego przy eksploatacji programów

Działanie systemu operacyjnego oraz wykonywane przez niego funkcje w trakcie realizacji poszczególnych programów przedstawiono dalej.

Działanie systemu operacyjnego maszyny NCR 315 dla cyklu programów

Rys. 3. Działanie systemu operacyjnego maszyny NCR 315 dla cyklu programów



☐ Wczytanie do pamięci programu SUPERVISOR za pomocą BOOT STREP znajdującego się na karcie dziurkowanej

- Wczytanie przez program SUPERVISOR programu A
- Wykonanie w maszynie programu A
- Program przez użyte makrorozkazy wykorzystuje program KERNEL, EXTREMITY, RESCUE
- Ostatni rozkaz (FINISCH) programu A wczytuje do pamięci programu SUPERVISOR
- Program SUPERVISOR wczytuje do pamięci nagłówek następnego programu i na podstawie informacji umieszczonych tam przez programistę za pomocą programu LIBRARIAN ustala, czy program powinien być uruchomiony, czy nie. Jeśli nie — wpisuje się na pulpicie maszyny, że program ten nie jest uruchamiany i przechodzi do wczytania nagłówka następnego programu
- Ustalenie, czy program B ma być uruchomiony
- Wczytanie do pamięci programu B
- Wykonanie programu B
- Wczytanie programu SUPERVISOR
- Ustalenie, czy program C ma być uruchomiony
- Wczytanie do pamięci programu C
- Wczytanie programu C
- Wczytanie do pamięci programu SUPERVISOR
- Ustalenie, czy program D ma być uruchomiony
- Wczytanie programu D
- Wykonanie programu D

Wczytanie programu SUPERVISOR

- Ustalenie, czy jest następny program, jeśli **nie**, umożliwienie operatorowi wywołania następnego programu według jego uznania przez podanie z pulpitu sterującego nazwy programu.

W przedstawionym automatycznym przebiegu realizacji cyklu programów, czynności oznaczone kwadratem wykonuje operator. Czynności oznaczone kółkiem wykonują bądź programy systemu operacyjnego, bądź programy użytkowe bez udziału operatora. Jak z tego widać, programy systemowe umożliwiają stworzenie dowolnych układów programów wchodzących w skład danego cyklu.

Poszczególne programy lub cykle programów mogą być przy przetwarzaniu automatycznie wczytywane do pamięci operacyjnej poprzez zastosowanie następujących rozwiązań, przewidzianych w programach systemowych:

- datowanie programów — w programie SUPERVISOR znajduje się trzyletni kalendarz. Na podstawie informacji zawartych w nagłówku programu, w oparciu o posiadany kalendarz, program SUPERVISOR może ustalić, czy w danym dniu program powinien być eksploatowany
- tworzenie ciągów programów — przy budowaniu biblioteki za pomocą programu LIBRARIAN można w nagłówku każdego programu podać nazwę następnego programu, który będzie wczytywany przez program SUPERVISOR

- datowanie zbiorów — podobnie, jak programy mogą być datowane również zbiory, tzn. można ustalić, ile dni dany zbiór jest ważny i nie może ulec zniszczeniu. Przeprowadzona przez EXTREMITY kontrola ważności zbioru może określić warunki przebiegu danego programu lub stwierdzić ważność określonego zbioru. Wykorzystywane jest to przede wszystkim dla zabezpieczenia przed pomyłkowym zapisaniem innych danych na ważnym zbiorze znajdującym się na zasobniku CRAM lub dla modyfikowania w programie użytkowym informacji wykorzystywanych przez system operacyjny:

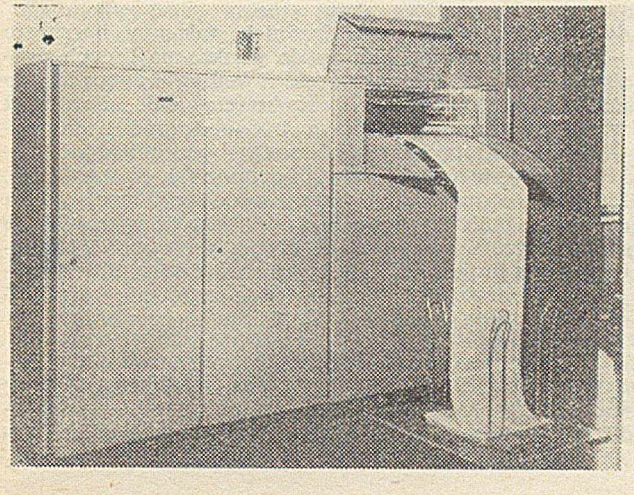
- zmiana nazwy następnego programu, np. gdy wiadomo, że w zależności od wyników przetwarzania ustalonych w programie mogą być uruchomione różne programy; po zakończeniu programu można zmodyfikować również nazwę następnego programu.

W pracach resortu finansów początkowo bardzo często występowały skomplikowane programy spełniające wiele różnych funkcji w zależności od ustaleń zewnętrznych, które operator musiał wprowadzić poprzez pulpit sterujący w trakcie wykonywania programu. Tego typu programy były bardzo uciążliwe w uruchamianiu, ponieważ często drobny błąd operatora, nie zawsze nawet możliwy do sprawdzenia, mógł spowodować błędne przetwarzanie długich cykli programów.

Ostatnio opracowuje się programy przewidujące już znacznie mniej interwencji operatora maszyny.

Wszystkie programy użytkowe, a także i operatorskie, są przechowywane w bibliotekach umieszczonych w zasobnikach CRAM. W jednym zasobniku można umieścić do 110 programów. Dostęp do programu jest natychmiastowy z odpowiedniej karty magnetycznej i bezbłędne wczytanie programu do pamięci operacyjnej nie przekracza 5 sekund. Wszystkie te ułatwienia umożliwiają tworzenie tzw. mutacji programów, tzn. tworzenie programów, które uruchamiane są w określonych warunkach i które jak gdyby same zabezpieczają się przed różnego rodzaju błędami operatora, gdyż nie wymagają żadnych informacji z pulpitu sterującego. Stwarza to warunki do maksymalnego zautomatyzowania poszczególnych prac. Możliwości takie trzeba jednak stworzyć już przy projektowaniu założeń do programów, tzn. przed rozpoczęciem opracowywania tych programów.

W warunkach Centrum Elektronicznego NBP, gdzie prace w cyklu dziennym pochłaniają dużo czasu maszyny, możliwość automatycznego przetwarzania stwarza warunki do maksymalnego wykorzystania czasu tej maszyny. W innym przypadku, gdyby programy przechowywane były na papierowych nośnikach informacji i każdorazowe wczytanie programu do pamięci uzależnione byłoby od decyzji operatora, czas niektórych prac wydłużyłby się bardzo poważnie, a w wielu przypadkach niemożliwe byłoby takie zorganizowanie pracy na maszynie, aby mogła być ona wykonywana wyłącznie przez operatorów.



Operatorzy ci nie znają założeń do poszczególnych prac i programów, a działanie swoje opierają na uproszczonej instrukcji obsługi, której podstawę stanowi tzw. organigram. Przedstawia on w sposób graficzny kolejność wykonania programów i ewentualnie wskazówkę, w którym momencie niezbędna jest interwencja operatora do zmiany zasobników CRAM i papieru w drukarce lub uruchomienia od nowa jakiegoś cyklu programów.

Tak więc przy pracach w cyklu dziennym, najważniejszą sprawą staje się odpowiednio sprawna organizacja automatycznego przetwarzania danych oraz maksymalna eliminacja interwencji operatora.

Nie mniej istotną sprawą, która występuje zwłaszcza przy pracach eksploatowanych okresowo, jest możliwość automatycznej eliminacji programów poprzez ich datowanie. Ułatwia to organizację pracy, uzależniając uruchomienie programu od uprzednio zaprogramowanych warunków. Stosując ten system możemy podawać określone daty ich uruchomienia lub wykonywać odpowiednie badania dat, a od wyników tych badań uzależniać uruchomienie programów. Ponadto system ten ma tę dobrą stronę, że wiele skomplikowanych przypadków rozwiązuje maszyna elektroniczna w oparciu o określony i dokładnie sprawdzony układ rozkazów dla systemu operacyjnego, uniezależniając wykonanie pracy od zawodnej pamięci pracowników. Bardzo często nieuzasadnione uruchomienie programu może zniekształcić wyniki całej pracy, co przy organizacji przetwarzania w Centrum Elektronicznym NBP bez udziału programistów lub innych pracowników znających założenia tej pracy mogłoby spowodować duże straty czasu maszyny, a nawet mogłyby ulec zniszczeniu lub zniekształceniu zbiory, które przechowywane są przez kilka miesięcy czy lat i których odtworzenie byłoby bardzo pracochłonne i nieraz nawet niemożliwe.

Zasady automatycznego przetwarzania bardziej obrazowo przedstawia wykaz czynności, jakie operator i inni pracownicy wykonują przy obsłudze pracy wykonywanej w cyklu dziennym, dotyczącej księgowości dla kilku oddziałów miejskich NBP. Czynności te przebiegają w sposób następujący:

1. Założenie właściwych zasobników CRAM na odpowiednie jednostki

2. Wywołanie pierwszego programu z cyklu „wejściowego"

3. Obsługa czytnika przy czytaniu taśmy dziurkowanej z danymi dziennymi

4. Po wykonaniu programów cyklu „wejściowego" odesłanie tabulogramów kontrolnych do Wydziału Kontroli Centrum i rozpoczęcie innej pracy

5. Po dostarczeniu przez Wydział Kontroli Centrum korekt na kartach dziurkowanych, powrót do obsługi tej pracy. Wywołanie pierwszego programu następnego cyklu. Obsługa czytnika kart dziurkowanych przy wczytaniu korekt

6. Obserwowanie prawidłowości przebiegu poszczególnych programów. Ewentualne podawanie z pulpitu określonych informacji, jeżeli program tego wymaga.

7. Zakończenie pracy dziennej. Przesłanie tabulogramów wynikowych do Wydziału Kontroli Centrum. Praca ta składa się w cyklu dziennym z minimum 25, a w określonych dniach i z większej liczby programów.

W swojej pracy operator maszyny z zasady posługuje się jedynie organigramem. W organigramie podaje się w formie odpowiednich symboli:

- nazwy programów
- z jakich zbiorów danych korzysta program
- czy wczytuje taśmę, czy karty drukowane

- czy drukuje tabulogram i w ilu egzemplarzach
- czy następny program jest wywoływany automatycznie
- czy powiązanie pomiędzy sąsiednimi programami jest bezpośrednie, czy też uzależnione od spełnienia określonych warunków badanych przez system operacyjny lub operatora.

W ten sposób operator otrzymuje wszystkie niezbędne informacje dla eksploatacji pracy na maszynie. Oczywiście zakłada się, że wszystkie programy są sprawdzone i nie zawierają błędów, ponieważ operator, który eksploatuje wiele różnych prac o różnorodnej tematyce, nie ma obowiązku znać merytorycznych założeń określonej pracy, a jedynie jej organizację eksploatacji na maszynie.

Przedstawione tu ogólne założenia systemu oraz zasady organizowania i programowania prac na maszynie wskazują, że posiadanie dobrze zorganizowanego systemu operacyjnego oraz umiejętne wykorzystanie jego możliwości stwarza warunki do intensywnej eksploatacji maszyny cyfrowej.

Jedną z zalet systemu operacyjnego NCR 315 jest zajmowanie stosunkowo niewielkiego obszaru pamięci operacyjnej (w zainstalowanej w NBP maszynie 8%), ponieważ większość programów systemowych jest wykonywana bądź na zasadzie odrębnych programów, bądź na zasadzie techniki OVERLAY [1]. Taka organizacja systemu operacyjnego jest bardzo wygodna, gdyż na zasobnikach kart magnetycznych są umieszczane tylko te programy systemowe, które mają być bezpośrednio wykorzystywane. Należy tu zaznaczyć, że w przypadku gdy w zasobniku kart magnetycznych mieszczą się równocześnie programy i zbiory danych, wówczas przez system operacyjny zajęte są jedynie cztery pierwsze karty, a więc około 2% całkowitej pojemności zasobnika.

W systemie operacyjnym NCR 315 można doszukać się pewnej analogii z systemami operacyjnymi maszyn III generacji. Wydaje się, że system NCR 315 jest wprawdzie systemem mniej uniwersalnym, ale za to bardzo oszczędnym, ponieważ zajmuje stosunkowo niewiele miejsca w pamięci operacyjnej.

Można podać jako ciekawostkę, że na maszynie NCR 315, która jest typową maszyną jednoprogramową, istnieją niekiedy możliwości jednoczesnego uruchomienia dwóch programów, z których jeden jest standardowym programem drukowania. W takim zestawieniu program zasadniczy wykorzystuje rezerwy czasowe, jakie powstają przy drukowaniu na szybkiej drukarce wierszowej. Program drukowania może być programem ciągłym, tzn. może działać w trakcie różnych zmieniających się programów aż do zakończenia operacji drukowania.

Oprócz tego maszyna NCR 315 posiada duże możliwości programowego zorganizowania podziału czasu jednostek peryferyjnych *(Time-Sharing)*, co jednak wymaga odpowiedniego zaprogramowania.

Na zakończenie niniejszego artykułu warto chyba podać, że większość rozwiązań dotyczących obsługi operatorskiej wprowadzano w Centrum Elektronicznym NBP sukcesywnie w wyniku wielu dyskusji. Głównym celem tych zamierzeń było zawsze jak najbardziej oszczędne wykorzystanie czasu maszyny w sytuacji bardzo dużego zapotrzebowania, zgłaszanego przez użytkowników resortu finansów, a jednocześnie ograniczonych do trzech zmian możliwości jednej maszyny cyfrowej.

---

[1]) Tzn. że odpowiedni program wczytywany jest do odpowiedniego obszaru pamięci wewnętrznej po uprzednim zabezpieczeniu przed zniszczeniem poprzedniej jego zawartości.

JERZY WOJCIECHOWSKI
Politechnika Gdańska
Instytut Technologii
Budowy Maszyn

ŁUDMIŁA ZAWADZKA
Politechnika Gdańska
Instytut Technologii
Budowy Maszyn

621.9—503.55:681.322.06

APT — system automatycznego programowania technologii na obrabiarki sterowane liczbowo

Opisano amerykański system automatycznego programowania w języku APT operacji technologicznych na obrabiarki sterowane liczbowo. Podano przykład programu.

Burzliwy rozwój sterowania liczbowego obrabiarek spowodował opracowanie całego szeregu typów sterowania oraz metod przygotowywania do nich programów. Cechą charakterystyczną układu sterowania obrabiarki jest konieczność przygotowywania programu obróbki dla każdej obrabianej części. Proces przygotowania programu obróbki części o nieskomplikowanych kształtach jest stosunkowo prosty, a wyznaczanie punktów, pomiędzy którymi porusza się narzędzie, jest szybkie i może być dokonane ręcznie. Natomiast zaprogramowanie ruchu narzędzia dla części bardziej skomplikowanej polega na obliczeniu współrzędnych dużej liczby punktów pośrednich, tak aby narzędzie — poruszając się od punktu do punktu po liniach prostych — mogło obrobić część według wymaganego kształtu i z założoną dokładnością [4]. Ponieważ wszystkie te punkty muszą być obliczone i zakodowane na taśmie sterującej, ręczne przygotowanie takiej taśmy byłoby niezwykle żmudne i czasochłonne. Zadanie to powierza się maszynie cyfrowej, która dokonuje obliczeń współrzędnych punktów trajektorii ruchu narzędzia oraz optymalnych prędkości jego ruchu; otrzymujemy więc dane wejściowe dla interpolatora obrabiarki — dane, które zapewnią osiągnięcie założonego kształtu i dokładności przedmiotu.

Złożoność zagadnień i konieczność wykorzystania maszyn cyfrowych stwarza potrzebę ścisłej współpracy biura technologicznego z ośrodkiem maszyn matematycznych w celu szybkiego, prawidłowego i opłacalnego przygotowania programów obróbki części.

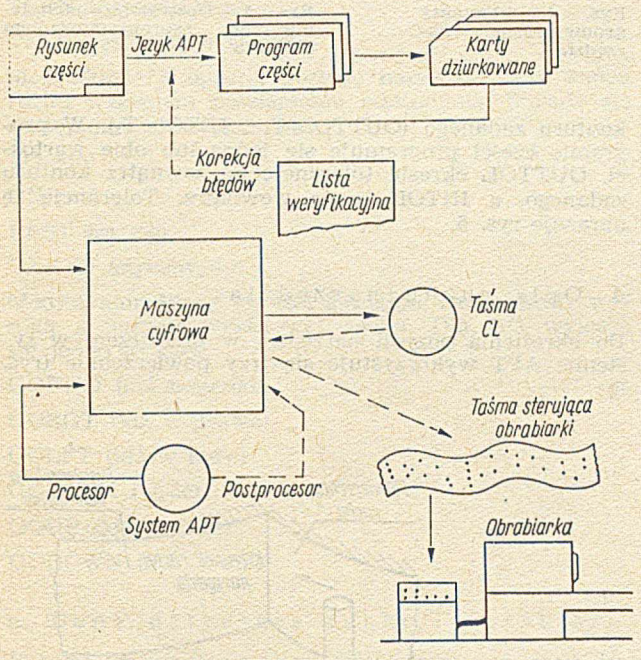
W chwili obecnej istnieje szereg systemów automatycznego programowania obróbki części przystosowanych do maszyn cyfrowych. Programowanie to odbywa się w oparciu o dane dotyczące geometrii części, jej dokładności, sposobu obróbki, warunków obróbki oraz specyfiki układu sterowania obrabiarki [1].

Jednym z systemów automatycznego programowania jest amerykański system APT dostosowany do programowania obróbki na obrabiarki ze sterowaniem kształtowym, punktowym oraz odcinkowym.

APT, który jest wspólnym językiem maszyn cyfrowych powstał w Laboratorium Fizyki Stosowanej Uniwersytetu John HOPKINS, wykorzystuje symbolikę Łukaszewicza. Symbolika ta została też zastosowana przez dr Bluma w Laboratorium Badawczym Marynarki w WHITEOAK MARYLAND oraz przez szereg innych ośrodków [2].

Opis systemu

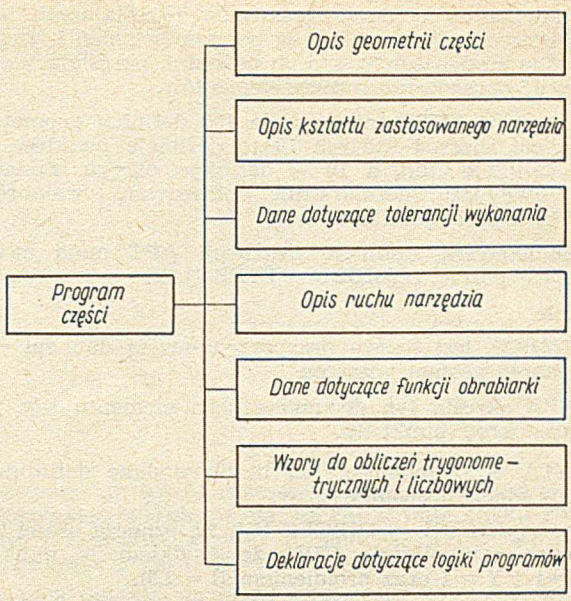
W celu zobrazowania całości systemu podano na rys. 1 kolejność symbolicznego programowania za pomocą APT [3].



Rys. 1. Kolejność symbolicznego programowania za pomocą APT

Technolog, który stosuje język APT do programowania procesu obróbki części, nie musi zasadniczo znać właściwości maszyny cyfrowej, która będzie realizować napisane przez niego programy. Może więc skupić swoją uwagę całkowicie na procesie obróbki, który ma być zaprogramowany. Pierwszym etapem przygotowania taśmy sterującej dla obrabiarki jest więc napisanie programu części w języku APT. Język ten składa się z około 300 stylizowanych angielskich słów o budowie łatwej do przyswojenia. Opierając się na rysunku części i narzuconych warunkach technicznych, inżynier-technolog pisze program, który składa się z szeregu elementów (rys. 2).

Na tym kończy się zadanie biura technologicznego, a rozpoczyna praca ośrodka maszyn matematycznych. Kiedy program części jest już kompletny, zostaje on wyperforowany na 80-kolumnowych kartach i wprowadzony do maszyny cyfrowej, która kompiluje program APT (p. rys. 1) [5].



Rys. 2. Elementy programu części

Maszyna cyfrowa tłumaczy najpierw instrukcje APT na zbiór podstawowych instrukcji maszynowych. W tym czasie program części jest sprawdzany na bieżąco, czy nie pominięto jakiegoś jego elementu oraz, czy nie ma błędów syntaktycznych. Jeśli zostaną one znalezione, cała ich lista zostaje drukowana, aby umożliwić jak najszybszą korektę programu. Gdy nie zostały ujawnione żadne błędy, program części jest kompilowany na program maszynowy, który wykona wszystkie niezbędne obliczenia. Po kompilacji przetłumaczony program wchodzi do maszyny cyfrowej, gdzie w jednostce centralnej ustalane są ogólne rozwiązania, oparte na programie części.

Wyniki przetwarzania w postaci zbiorów współrzędnych punktów zapisane są na taśmie magnetycznej. Punkty wyznaczane są w taki sposób, aby środek narzędzia poruszał się między nimi po odcinkach prostych tak dobranych, aby została obrobiona część zgodnie z wymaganiami programu części. Taśma ta jest zwana taśmą CL (*cutter location* — pozycja narzędzia).

Dane maszynowe znajdujące się na taśmie CL są zawsze tej samej postaci i muszą być przetwarzane przez kolejną część systemu tzw. „*postprocessor*" na formę dostosowaną do systemu sterowania danej obrabiarki. Dostarczanie postprocessora do maszyny cyfrowej jest stosunkowo łatwe i, jeśli jakiś postprocessor jest przygotowany do danej obrabiarki, nie wymaga się żadnego dodatkowego programowania.

Efektem końcowym jest taśma sterująca dla obrabiarki z wypisanym dla niej programem w kodzie maszyny.

Przygotowanie danych wejściowych dla maszyny cyfrowej

Aby programista ośrodka maszyn matematycznych mógł właściwie wykorzystać program opracowany w biurze technologicznym, powinien on znać zarówno elementy, z jakich składa się ten program (patrz rys. 2), jak i język APT wykorzystany do jego zapisu.

W dalszej części artykułu omówiono w skrócie te elementy.

a. Opis geometrii części

W pierwszej kolejności część dzieli się na zbiór określonych geometrycznie elementów. W układzie dwuwymiarowym elementy te mogą być odcinkami, okręgami, elipsami itp. Niektóre z nich mogą być zde-

finiowane za pomocą parametrów określających bezpośrednio kształt, np. okrąg poprzez promień i współrzędne środka lub w sposób pośredni, określający styczność okręgów do innych elementów.

W języku APT istnieje około 100 definicji geometrycznych, spośród których 10 to definicje punktów, 11 — definicje linii, a 10 — definicje okręgu. Ponadto możliwe jest zdefiniowanie płaszczyzn, wektorów, elips, hiperbol, walców, kul.

Geometryczne definicje instrukcji APT mają następującą postać: SYMBOL = TYPE/DATA

gdzie

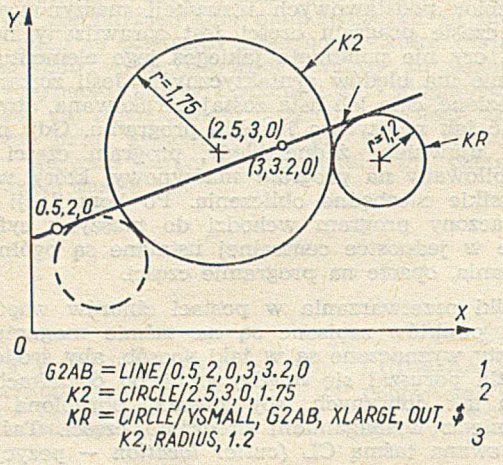
SYMBOL jest nazwą przyporządkowaną danemu elementowi geometrycznemu

TYPE określa typ geometrycznego elementu tak, jak linia, okręg, punkt itp.

DATA — symboliczne lub liczbowe dane definiujące dany element geometryczny.

Np. zapis CIR1 = CIRCLE (1,1, 2,5 oznacza okrąg nazwany symbolicznie CIR1 ze środkiem w punkcie X =1 i Y = 1 oraz promieniem R = 2,5).

Na rys. 3 pokazano przykłady definicji geometrycznych. Zdefiniowana przez dwa punkty linia otrzymu-



Rys. 3. Przykład definicji geometrycznych dwóch okręgów i prostej

je symbol G2AB. Współrzędne X, Y, Z obu punktów występują po słowie LINE (LINIA). Następna deklaracja — okrąg — jest zdefiniowana symbolem K2. Po słowie CIRCLE podano współrzędne środka okręgu i jego promień. W trzeciej linii zdefiniowano okrąg KR, przy czym widoczne są tu zalety zastosowania języka symbolicznego. Okrąg KR można skonstruować wykorzystując dwie uprzednie definicje i podany promień. Leży on pod linią G2AB, tj. po stronie skierowanej ku mniejszej współrzędnej Y. Dlatego zapisano ySMALL (y - małe). KR ma być ponadto styczny do okręgu K2 po jego zewnętrznej stronie, co zaznaczono słowem OUT. XLARGE oznacza większą współrzędną X. Na końcu definicji zapisano słowo RADIUS (PROMIEŃ) i jego wartość.

b. Opis kształtu zastosowanego narzędzia

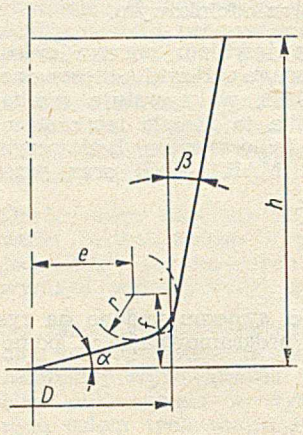
System APT przyjmuje obrotową symetrię narzędzia. Deklaracja dotycząca kształtu narzędzia ma postać ogólną typu:

CUTTER/D, *r*, *e*, *f*, *α*, *β*, *h*

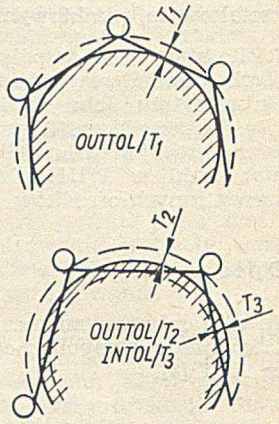
Znaczenie poszczególnych parametrów określa rys. 4.

c. Dane dotyczące tolerancji wykonania

W celu odpowiedniego zagęszczenia punktami konturu obrabianego w programie umieszcza się dane dotyczące dozwolonego odchylenia od zaprogramowanego konturu. Tolerancję określa się po obu stronach



Rys. 4. Elementy geometryczne narzędzia

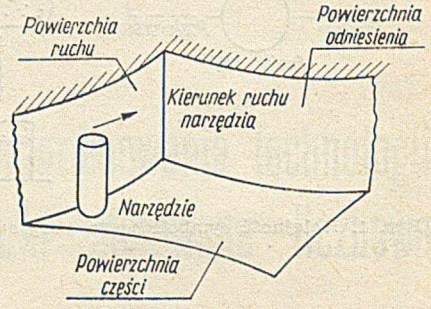


Rys. 5. Dozwolone odchylenie od zaprogramowanego konturu

konturu zadanego (OUTTOL/T_1 i INTOL/T_2). W programie części programuje się jedną lub obie wartości. OUTTOL określa tolerancję na zewnątrz konturu zadanego, a INTOL — do wewnątrz. Tolerancję tę obrazuje rys. 5.

d. Opis ruchu narzędzia

Do określenia pozycji narzędzia w przestrzeni w systemie APT wykorzystuje się trzy powierzchnie (rys. 6):



Rys. 6. Sposób określania pozycji narzędzia w przestrzeni

1. Powierzchnia ruchu narzędzia oznaczona przez DS *(drive surface)*

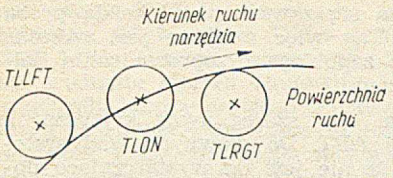
2. Powierzchnia przedmiotu PS *(part surface)*

3. Powierzchnia odniesienia CS *(check surface)*.

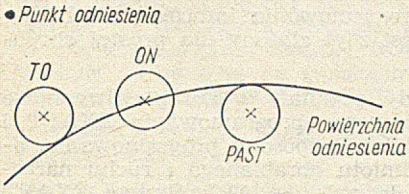
Narzędzie porusza się wzdłuż powierzchni ruchu. Dla frezarki pionowej, gdzie narzędzie jest prostopadłe do płaszczyzny X Y stołu, powierzchnia ruchu odnosi się do konturu i daje współrzędne X, Y każdego punktu drogi środka narzędzia.

Powierzchnia części jest powierzchnią, po której porusza się punkt końcowy osi narzędzia. Dla frezarki pionowej powierzchnia ta daje współrzędne poszczególnych punktów drogi.

Powierzchnia odniesienia ogranicza ruch narzędzia poruszającego się po elemencie konturu.



Rys. 7. Położenie narzędzia względem powierzchni ruchu podczas jego przesuwania się



Rys. 8. Położenie narzędzia względem powierzchni odniesienia przy końcu ruchu

Do zapisu deklaracji ruchu stosuje się:

Modyfikator 1/GO *(drive surface)*, służący do zdefiniowania pozycji narzędzia w odniesieniu do powierzchni ruchu DS.

Stosuje się następujące słowa (rys. 7):

TLRGT (narzędzie w prawo)

TLON (narzędzie na)

TLLFT (narzędzie w lewo)

Modyfikator 2 opisuje pozycję narzędzia przy końcu ruchu względem powierzchni odniesienia. Stosuje się następujące słowa (rys. 8):

TO (do)

ON (na)

PAST (za, po)

TANTO (styczny do)

Modyfikator, który ma być aktualnie zastosowany wynika z pozycji wyjściowej ruchu. Dla GO wyłącza się jedno ze słów:

GOFWD (idź naprzód)

GORGT (idź w prawo)

GOLFT (idź w lewo)

GOBACK (wróć)

GOUP (idź w górę)

GODOWN (idź w dół)

e. Dane dotyczące funkcji obrabiarki

Programista programujący części może umieścić deklarację realizacji określonych funkcji obrabiarki pomiędzy deklaracjami ruchu narzędzia:

Np. określenie wielkości posuwu

FEDRAT/50

FEDRAT/RAPID

● Określenie szybkości obrotowej wrzeciona

SPINDL/3000, CLW

● Chłodziwo

COOLNT/MIST

COOLNT/OFF

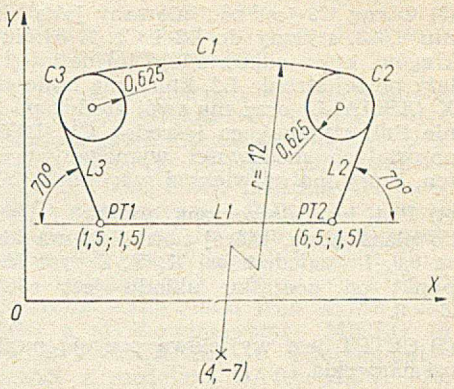
Stosuje się również deklarację MACHIN/... do zaznaczenia rodzaju obrabiarki, dla której pisany jest program.

f. Obliczenia maszyny cyfrowej

Za pomocą wzorów matematycznych można obliczyć określone wartości w ramach systemu APT. Do zapisu wzorów używa się języka FORTRAN. Język APT obejmuje deklarację arytmetyczną o ogólnej postaci:

zmienna = wyrażenie matematyczne

Rys. 9. Rysunek krzywki



Deklaracja ta ustala rozkaz dla maszyny cyfrowej do oszacowania wyrażenia po prawej stronie znaku równości i zapisania wyniku w pamięci. Wzory zapisuje się również w postaci symbolicznej, np.

arc tg *b* oznacza się przez ATANF(B)

$\sqrt{x}$ oznacza się przez SORTF(X)

exp x oznacza się przez EXP(X)

$\log_e x$ oznacza się przez LOGF(X)

W celu oznaczenia działań arytmetycznych stosuje się następujące znaki : +, —, *, /, π odpowiadające dodawaniu, odejmowaniu, mnożeniu, dzieleniu i potęgowaniu.

Przykład programowania części w języku APT [5]

Na rys. 9 pokazano prostą część, dla której mamy wypisać program. Dalej podano kompletny program, zapisany w języku APT:

L1 = LINE/PT1 = POINT 1.5, 1.5, PT2 = POINT/6.5, 1.5 (1)

L2 = LINE/PT2, ATANGL, 70 (2)

L3 = LINE/PT1, ATANGL, 110

C1 = CIRCLE/4,—7, 12

C2 = CIRCLE /XSMALL, L2, YIARGE, INTAN, C1 RADIUS, 0.625 (3)

C3 = CIRCLE /XSMALL, L3,YLARGE, INTAN, C1 RADIUS, 0.625

PL1 = PLANE/0, 0, 1, 0 (4)

SETPT = POINT/0, 0, 3 (5)

CUTTER/0,5 (6)

FROM/SETPT

GO/TO, L1, TO, PL1 (7)

GORGT/L1, PAST, L2 (8)

GOLFT/L2, TANTO, C2 (9)

GOFWD/C2 (10)

GOFWD/C1

GOFWD/C3

GOFWD/L3, PAST, L1

GOTO/SETPT

FINI

Ważniejsze punkty programu zaznaczono liczbami w nawiasach okrągłych i należy je rozumieć następująco:

(1) L1 jest linią przechodzącą przez punkty PT1 i PT2 określone w definicji. Nie podano wartości współrzędnych „Z", gdyż równe są zeru.

(2) Linia L2 jest zdefiniowana jako linia przechodząca przez punkt PT2 i tworząca kąt 70° z dodatnią osią odciętych.

(3) Okrąg C2 jest zdefiniowany jako okrąg o promieniu 0,625, styczny do L2 i C1. Z ośmiu możliwych okręgów, które odpowiadają tej definicji, żądany okręg leży po tej stronie L2, która ma mniejszą współrzędną X (XSMALL) oraz ma swój środek po tej samej stronie linii stycznej, co i środek C1 (INTAN). Z dwóch pozostałych możliwości wymaganym okręgiem jest ten, który ma największą wartość Y (YLARGE).

(4) PL1 jest płaszczyzną obróbki i jest określona jako płaszczyzna, której normalna ma kierunek cosinusa 0,0, 1 względem osi X, Y, Z oraz której najbliższy punkt od początku układu jest równy zeru, tzn. Z = 0.

(5) SETPT jest wyjściową pozycją punktu końcowego osi narzędzia.

(6) Średnica narzędzia została określona jako 0.5.

(7) Obrabiarka otrzymuje rozkaz poruszania się z pozycji wyjściowej do pozycji, w której będzie dotykać zarówno L1, jak i PL1. Aby osiągnąć tę pozycję, zostaje wybrany możliwie najbliższy punkt SETPT.

(8) Obrabiarka otrzymuje rozkaz ruchu w prawo i obróbki wzdłuż L1 aż do punktu przecięcia z L2.

(9) Obrabiarka otrzymuje rozkaz ruchu w lewo i obróbki wzdłuż L2 aż do punktu, w którym ostrze będzie styczne do C2.

(10) Narzędzie otrzymuje rozkaz ruchu naprzód wzdłuż C2.

Nie podano żadnego modyfikatora, mówiącego o zakończeniu obróbki wzdłuż C2. Informacja ta zawarta jest w następnej instrukcji, która daje rozkaz ruchu wzdłuż C1. Obrabiarka zakończy obróbkę wzdłuż C2 wtedy, gdy narzędzie napotka C1.

Uwagi końcowe

Opisany system APT jest unowocześnioną wersją powstałego w roku 1961 systemu APT III. Do głównych zalet systemu APT należy to, że można łatwo rozszerzać jego możliwości. Poza tym kompilacja programu części odbywa się zawsze przed każdą próbą jego przetwarzania. Tak więc możliwe jest wczesne wykrycie błędów, co zapobiega przeprowadzaniu dalszych błędnych obliczeń.

W Europie wysuwa się szereg uwag krytycznych przeciwko systemowi APT. Do głównych wad systemu APT należy to, że nie jest on wystarczająco dostosowany do obrabiarek ze sterowaniem punktowym, odcinkowym, do programowania informacji technologicznych oraz że jest zbyt złożony dla prostej obróbki dwuwymiarowej.

Aby usunąć te wady, powstały w Europie dwa nowe systemy automatycznego programowania EXAPT i 2CL. W celu normalizacji sposobu programowania, opis geometrii przedmiotu obrabianego i ruchu narzędzia jest podobny do opisu w APT. System EXAPT posiada ponadto możliwości podejmowania decyzji technologicznych podczas obliczeń na maszynie cyfrowej odnośnie do wyboru wielkości posuwów, szybkości, drogi oraz rodzaju narzędzia.

Wzrastające gwałtownie zastosowanie obrabiarek sterowanych programowo stwarza potrzebę opracowania znormalizowanego i uniwersalnego systemu automatycznego programowania umożliwiającego szybkie i precyzyjne opracowanie programu obróbki każdej części w dowolnym ośrodku obliczeniowym. Wyrazem tej potrzeby stała się polska próba systemu automatycznego programowania — WTP. System ten został opracowany w Instytucie Elektrotechniki (patrz [1]).

BIBLIOGRAFIA

[1] Adriana Gralak, Wiesław Gralak — „Systemy przygotowywania informacji do sterowania liczbowego obrabiarek" — „Mechanik", nr 5/1969

[2] R. P. Rich — „APT, a Common Computer Language" — Annual Review in Automatic Programming, Pergamon Press 1961

[3] K. Janzen — „NC users guide to the international Computer programmes — APT", Metalworking Production, 29 Nov. 1967

[4] James J. Childs — „Principles of Numerical Control", Copyright C 1955 by Industrial Press, New York N. Y.

[5] „APT — The New System", Data Processing, nr 11/12/1969.

TOMASZ PAWLAK
Instytut Maszyn Matematycznych
Warszawa

COMPUTER 70 — Światowe spotkanie producentów i użytkowników maszyn cyfrowych

Londyn, 5 — 9. X. 1970

1. Ogólna charakterystyka Wystawy

Wystawa COMPUTER 70 zorganizowana przez Brytyjskie Stowarzyszenie Handlu Sprzętem Orgatechniki (Business Equipment Trade Association) w wielkiej hali londyńskiej OLYMPII zgromadziła około 220 wystawców z 11 krajów (Belgia, Izrael, Francja, Holandia, Kanada, NRD, NRF, Polska, USA, Węgry, Włochy).

Ze względu na miejsce organizacji Wystawy, główną część wystawców stanowiły firmy brytyjskie. Wrażenie to potęgowało się tym bardziej, że znane światowe firmy komputerowe, jak IBM, UNIVAC i inne były reprezentowane na Wystawie przez swoje oddziały działające w Wielkiej Brytanii. Udział w Wystawie firm francuskich i NRF-owskich był niewielki i z pewnością nie odzwierciedlał potencjału przemysłu komputerowego tych krajów. Tłumaczyć to można tym, że skoncentrowały się one na „konkurencyjnej" wystawie SICOB 70, która odbyła się tydzień wcześniej w Paryżu.

W wystawie COMPUTER 70 uczestniczyły prawie wszystkie firmy, które się liczą na rynku anglosaskim, z wyjątkiem BULL-GE i SIEMENSA. Druga co do wielkości firma światowa HONEYWELL-GE była obecna na Wystawie raczej symbolicznie. HONEYWELL wystąpił tylko z klawiaturami, zaś GE w ogóle nic nie wystawiał.

Polska była reprezentowana przez BHZ ELWRO, które przedstawiło kilka urządzeń zewnętrznych produkowanych w zakładach Zjednoczenia MERA: czyt-

nik taśmy CT-1001, dziurkarka taśmy D-102, pamięć bębnowa PB-7, bęben BW-8 i głowice magnetyczne GPT-3z do pamięci taśmowych. Urządzone skromnymi środkami stoisko stanowiło świadectwo istnienia przemysłu komputerowego w PRL i dawało obraz naszych możliwości. Jednakże w celu wejścia na rynek brytyjski konieczne są dla nas powiązania z firmami działającymi na tym terenie. Z drogi tej korzysta NRD, która formalnie nie uczestniczyła w Wystawie, jednakże jej techniczną obecność w postaci automatów obrachunkowych CELLATRON i maszyn SOEMTRON widać było na stoiskach kilku firm, które sprzedają te wyroby w Wielkiej Brytanii.

Na Wystawie przedstawiały swoje możliwości zarówno firmy–producenci:

- całych systemów komputerowych
- wybranych urządzeń komputerowych
- nośników informacji,

jak też i firmy usługowe zajmujące się:

- udzielaniem konsultacji co do możliwości i sposobu zastosowań komputerów w przedsiębiorstwie
- projektowaniem systemów API
- opracowywaniem specjalistycznego oprogramowania
- informacją techniczną, publikacjami i wydawnictwami książkowymi w zakresie zastosowań i budowy komputerów.

Jednakże ton nadawały firmy-producenci, które zajmowały ponad 70% powierzchni wystawowej.

Wystawa dawała obraz możliwości firm biorących w niej udział, stanowiła ofertę dla odbiorców-użytkowników komputerów, a także firm kompletujących systemy komputerowe. Chociaż nie miała ona charakteru targów, dała okazję do nawiązania szeregu kontaktów z klientami, które w przyszłości zwiększą obroty handlowe firm w niej uczestniczących.

2. Tendencje rozwojowe produkcji i zastosowań maszyn cyfrowych

Podstawową tendencją, która wystąpiła na COMPUTER 70 jest dążność do ułatwienia i potanienia dostępu do komputerów dla możliwie najszerszego kręgu użytkowników. Realizacja tej tendencji odbywa się dwoma sposobami:

a) na drodze zdalnego przetwarzania (teleprocessing-u), tj. budowy średnich i dużych systemów komputerowych z jednostkami centralnymi pracującymi w systemie podziału czasu (time sharing), do których podłącza się poprzez urządzenia transmisji danych – dużą ilość końcowych urządzeń abonenckich (terminals)

b) na drodze budowy minikomputerów.

Teleprocessing był podstawowym elementem pokazu dużych firm komputerowych; znajdujące się na ich stoiskach urządzenia końcowe podłączone były do maszyn zainstalowanych poza terenem Wystawy:

- IBM – komunikacja z maszynami 360/40 w Croydon oraz 360/50 i 360/65 w Londynie
- UNIVAC – komunikacja z maszynami 9400 i 1108 w Londynie
- BURROUGHS – zdalne przetwarzanie z pomocą komputera B 5500, pracującego w Londynie. Na stoisku demonstrowano możliwości komputera B 3500 w zakresie wieloprogramowości i wieloprocessing-u (maksymalna możliwość 256 programów jednocześnie).
- ICL – komunikacja z maszyną systemu 4/50 w Putney.

Wśród urządzeń końcowych w powszechnym użyciu są alfanumeryczne monitory ekranowe. Wytwarzają je zarówno wszystkie duże firmy produkujące systemy komputerowe, jak również i małe firmy o wąskim asortymencie produkcji. Na wystawie COMPUTER 70 alfanumeryczne monitory ekranowe były jednym z najliczniej reprezentowanych urządzeń.

Minikomputery jako alternatywne rozwiązanie taniego i łatwego dostępu do maszyny cyfrowej były proponowane klientom przez kilkanaście małych firm. Wśród nich czołową pozycję zajmuje amerykańska DIGITAL EQUIPMENT Co., która przedstawiła swoją znaną serię PDP z najnowszym PDP-8/E, a także PDP-15/30, przeznaczonym do obliczeń inżynierskich i sterowania w czasie rzeczywistym. Poprzez linię transmisji danych dostępne były z Wystawy PDP-12 i PDP-105. Pozostałymi wystawcami tego wyrobu były firmy: DIGICO z minikomputerami MICRO 16 i NORD 1, HEWLETT-PACKARD z minikomputerami 9100, VARIAN z modelami DATA 520 & 620, VIATRON z modelami System 21 i inne.

Eksponowane na COMPUTER 70 urządzenia przygotowania danych wskazują, że ten odcinek jest drugim kierunkiem intensywnego rozwoju systemów przetwarzania danych, który podąża za dokonanym w latach poprzednich rozwojem jednostek centralnych i urządzeń zewnętrznych maszyn cyfrowych. Wystawcy szeregu urządzeń przygotowania danych potwierdzali ten fakt, reklamując swój sprzęt jako rozszerzenie III generacji maszyn cyfrowych również i na odcinek przygotowania danych.

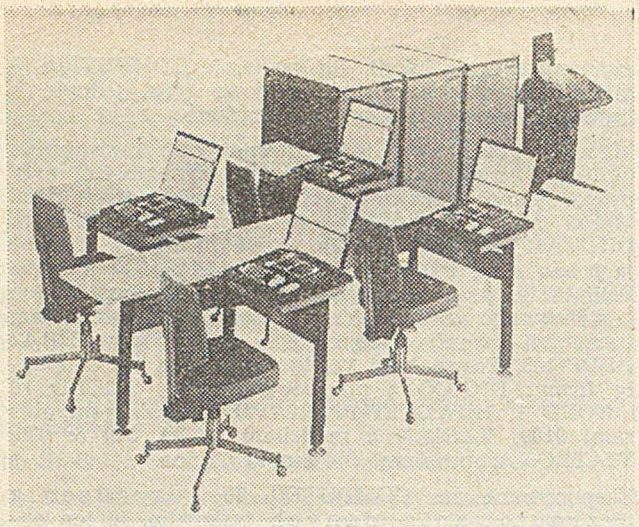
Podstawową cechą nowoczesnych urządzeń przygotowania danych staje się zastosowanie taśmy magnetycznej lub dysku magnetycznego jako wymiennego nośnika informacji, który po zarejestrowaniu danych może być bezpośrednio użyty w jednostce pamięci zewnętrznej i pozwala wprowadzać z dużą prędkością informację do komputera. Obserwacja reklamy urządzeń prowadzi do wniosków o ograniczeniu rozwoju systemów przygotowania i wprowadzania danych na kartach dziurkowanych oraz o stabilizacji rozwoju i zastosowań w zakresie systemów opartych na taśmie dziurkowanej. Spostrzeżenia te nie oznaczają z pewnością spadku produkcji i zastosowań kartowych urządzeń przygotowania i wprowadzania danych. Z jednej strony przemawia za dalszym ich stosowaniem wieloletnia tradycja i przyzwyczajenie użytkowników, posiadających już dotąd wielki park tych urządzeń, a od strony wytwórców przemawia za kontynuacją produkcji bogate jej oprzyrządowanie i zorganizowana szeroko sieć obsługi.

Jednakże w ogólnym poziomie produkcji urządzeń przygotowania danych następuje względny spadek udziału urządzeń opartych na stosowaniu kart i taśm dziurkowanych.

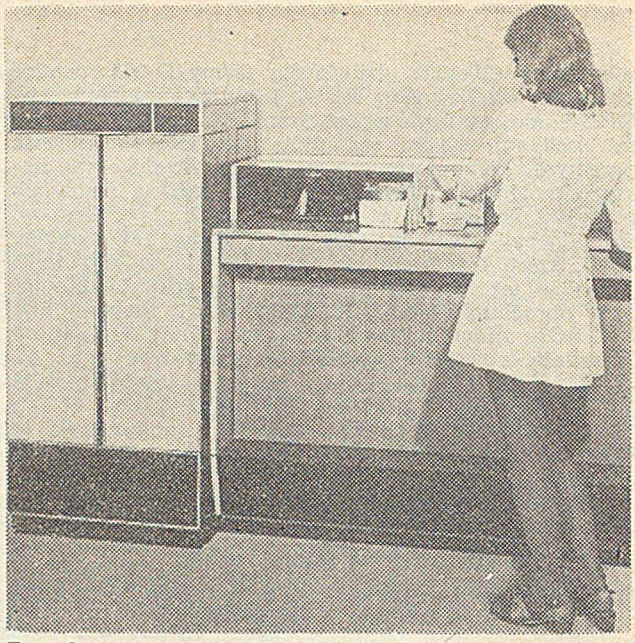
Urządzenia przygotowania danych na standardowej 1/2-calowej taśmie magnetycznej z jednym pulpitem wejściowym (klawiaturą) są produkowane już od kilku lat i posiadają szeroki krąg użytkowników. Prekursorem w ich produkcji, który w poważnym stopniu opanował rynek światowy, jest amerykańska firma MOHAWK DATA SYSTEMS (MDS).

Ostatni okres przyniósł pojawienie się szeregu rozwiązań systemów przygotowania danych z wieloma klawiaturami i urządzeniem sterującym, służącym do koordynowania wprowadzania z nich informacji na wspólną szpulę z taśmą magnetyczną lub na wspólny dysk wymienny. Urządzenia sterujące w takich systemach posiadają możliwości programowego sterowania i wszechstronnej kontroli wprowadzanych informacji.

Najelegantszym i najbardziej konsekwentnym rozwiązaniem wielowejściowego systemu przygotowania danych był system KEY-EDIT 100, przedstawiony przez kanadyjską firmę CONSOLIDATED COMPUTER SERVICES. Daje on możliwości podłączenia $8 \div 32$ klawiatur. Funkcję programowanego urządzenia sterującego realizuje w nim minikomputer PDP-8 z pamięcią ferrytową o pojemności 4 Kbajtów. Pamięcią pośrednią jest bęben o pojemności 320 Kbajtów. Wprowadzone z klawiatur informacje po zweryfikowaniu i wstępnym uporządkowaniu są rejestrowane



Rys. 1.



Rys. 2.

z pomocą wolnej pamięci taśmowej na standardowej 10 i 1/2 calowej szpuli z 1/2-calową taśmą magnetyczną lub z pomocą pamięci dyskowej na wymiennym standardowym 6-talerzowym pakiecie dysków magnetycznych. Na podkreślenie zasługuje fakt, że rozwiązanie systemu KEY-EDIT 100 (rys. 1) zostało w całości oparte na urządzeniach wytwarzanych przez różne firmy, tj. minikomputer PDP-8, pamięć bębnowa, pamięć dyskowa, wolna pamięć taśmowa oraz klawiatura.

9876543210S 24680135794 9753124680H | 83475

a)

9876543210H 2468013579E 9753124680KR 52731

b)

Zasługą firmy CONSOLIDATED COMPUTER SERVICES jest opracowanie koncepcji struktury systemu wraz z uzupełnieniem zakupionych urządzeń o niewielkie układy zabezpieczające ich współpracę i realizację funkcji systemowych przy jednoczesnym skierowaniu dużego wysiłku na opracowanie specjalizowanego oprogramowania dla tego systemu. Zalety systemu KEY-EDIT 100 wysoko ocenia firma ICL, która zakupiła licencję na ten system i propaguje jego stosowanie w systemach przetwarzania danych, opartych na maszynach serii 1900 oraz Systemu 4.

Wielowejściowe urządzenia przygotowania danych prezentowały także firmy:

COMPUTER MACHINERY Co	— System CMC 7
INTERSCAN DATA SYSTEMS	— System 2100
MOHAWK DATA SYSTEMS	— MDS 9000
REDIFON Ltd	— KEYCHECK

i inne

Wzrost liczby produkowanych i zainstalowanych komputerów, wykorzystujących pamięci masowe na taśmach i dyskach magnetycznych, a także rozwój zastosowań urządzeń przygotowania danych używających tych samych nośników informacji spowodowały istotny wzrost zapotrzebowania na taśmy i dyski magnetyczne. Zjawisko to stanowi ekonomiczne uzasadnienie podejmowania produkcji tych nośników informacji, pomimo koniecznych dużych nakładów finansowych na jej uruchomienie. Na Wystawie widać było wyraźnie wzrost liczby europejskich producentów pakietów dysków i taśm magnetycznych.

Wzrost produkcji i zastosowań optycznych czytników dokumentów, (rys. 2ab) drukowanych znakami o standardowym kształcie OCR A a) lub OCR B b) (według ISO) jako urządzeń do bezpośredniego wprowadzania danych z dokumentów pierwotnych do komputera — stanowi konkurencyjną (ale dopiero w przyszłości) linię rozwoju względem opisanych tutaj urządzeń przygotowania danych. Często są one stosowane do wprowadzania do maszyny cyfrowej wydruków z taśm papierowych produkowanych przez kasy rejestrujące bądź zwykłe elektromechaniczne sumatory. Optyczne czytniki dokumentów przedstawiały m. in. firmy: BURROUGHS, CONTROL DATA, DATA RECOGNITION Ltd, INTERSCAN, PLESSEY, RECOGNITION EQUIPMENT i VIATRON.

Mikrofilmowe urządzenia wyjściowe do komputerów pojawiły się w roku 1970. Na COMPUTER 70 pokazywały je:

- BURROUGHS — BCOM microfilmer
- DATAGRAPHIX Ltd — DATAGRAPHIX 4440
- KODAK — KOM 90 microfilmer
- MEMOREX — 1603 microfilmer

Atrakcyjność tych urządzeń polega na miniaturyzacji objętości, którą zajmuje na mikrofilmie informacja wyjściowa z maszyny cyfrowej w porównaniu do objętości, którą ta sama ilość informacji zajmuje na arkuszach drukarki wierszowej. Wadą tej techniki jest niemożliwość bezpośredniego odczytu informacji z mikrofilmu przez użytkownika maszyny cyfrowej. Być może jednak, w przyszłości ta niedogodność zostanie zmniejszona i mikrofilmowe urządzenia wyjściowe zastąpią powszechnie stosowane obecnie drukarki wierszowe.

3. Podsumowanie i wnioski

Przedstawione w tym artykule zasadnicze wrażenia nie powinny wywołać u Czytelnika uczucia, że na COMPUTER 70 nie było właściwie „pełnowymiarowych" komputerów. Owszem — były — chociaż liczba eksponowanych typów stanowi nieznaczny procent dostępnych na rynku europejskim i światowym. Większość z nich zbudowana jest ze standardowych układów scalonych, przy częściowym wprowadzeniu do nich układów o średniej (MSI), a nawet dużej skali integracji (LSI).

Firma IBM wystawiła mały komputer SYSTEM/3 (rys. 3), posiadający pamięć dyskową i wielofunkcyjną jednostkę na miniaturowych kartach dziurkowanych. W komputerze tym zastosowano szereg technologii wielkoscalonych. System 370 był obecny na stoisku IBM tylko w postaci prospektowej.

Z nowościami w dziedzinie komputerów małej i średniej mocy wystąpiły:

- CONTROL DATA — z komputerem CDC 1700
- NCR — z komputerem CENTURY 100
- RANK XEROX DATA SYSTEMS — z komputerem SIGMA 3
- SINGER-FRIDEN — z komputerem SYSTEM 10

Wśród nich interesujące jest rozwiązanie SIGMA 3 — uniwersalny komputer do obliczeń naukowych. Jego konfiguracja z monitorem alfanumerycznym zawiera dodatkowy subprocessor, który pozwala na realizację obliczeń prawie niezależnie od operacji wejściowo-wyjściowych poprzez monitor.

Ponadto na COMPUTER 70 pokazano dość liczną grupę urządzeń takich, jak:

a) automaty obrachunkowe (księgujące, fakturujące)

b) magnetyczne pamięci bębnowe, dyskowe i taśmowe

c) pamięci wewnętrzne komputerów oraz pamięci buforowe dla szeregu urządzeń zewnętrznych

d) klawiatury (dla monitorów ekranowych i urządzeń przygotowania danych)

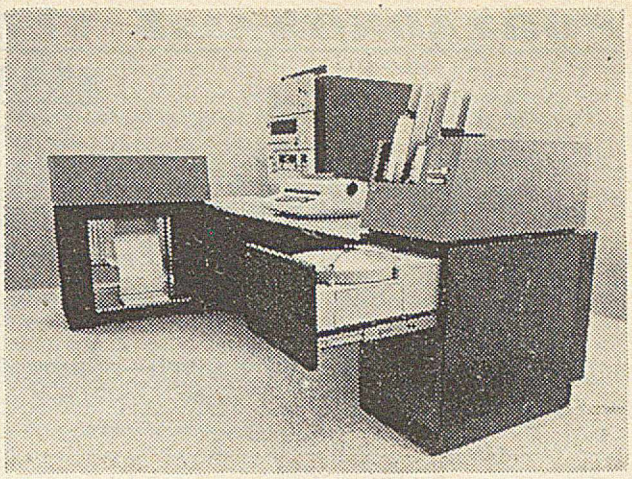
e) sprzęt pomocniczy (do wyposażenia ośrodków obliczeniowych).

W ogólnej ocenie można uznać, że wystawa COMPUTER 70 dała pełny przegląd systemów i urządzeń komputerowych dostępnych obecnie na rynku światowym.

Zaobserwowane i przedstawione światowe tendencje rozwojowe produkcji i zastosowań komputerów pozwalają sformułować kilka wycinkowych wniosków, dotyczących rozwoju technicznego systemów informatyki w kraju:

1. Należy dążyć do wprowadzenia do eksploatacji systemów informatyki, opartych na teleprocessingu.

W tym celu należy określić dziedziny gospodarki narodowej, rodzaje zastosowań komputerów, w których zastosowanie teleprocessingu powinno odbywać się w pierwszej kolejności jako najbardziej opłacalne z punktu widzenia kosztów wprowadzania kompleksowego przetwarzania informacji, jak i celowe z punktu widzenia wymagań funkcjonalnych dla systemu informacyjnego. Dlatego niezbędny jest równocześnie rozwój sieci transmisji danych i kompleksowe podejście



Rys. 3.

do krajowej produkcji i importu z innych krajów socjalistycznych do ich budowy.

2. Należy powiązać planowany rozwój krajowej produkcji minikomputerów z badaniami efektów ekonomicznych ich zastosowań w różnych dziedzinach w porównaniu z korzyściami zastosowań komputerów średniej wielkości (wraz z teleprocessingiem lub bez) i na tej podstawie ukierunkowywać wprowadzanie minikomputerów do gospodarki narodowej.

3. W zakresie urządzeń przygotowania danych, będących tradycyjnie u nas przedmiotem importu, na które zapotrzebowanie wzrastać będzie szybko ze wzrostem liczby instalowanych komputerów — należy nastawiać się głównie na import urządzeń, opartych na taśmie magnetycznej i dążyć do jak najszybszego uruchomienia w kraju produkcji takich urządzeń.

4. Należy dążyć do jak najszybszego zabezpieczenia przez krajową produkcję rosnącego zapotrzebowania na cyfrową taśmę magnetyczną jako podstawowego w niedługiej już przyszłości nośnika do wprowadzenia, zapamiętywania i przechowywania informacji.

Oczywiście powyższe wnioski mają jedynie charakter cząstkowy, gdyż uwypuklają tylko niektóre potrzeby rozwoju technicznego systemów informatyki na podstawie spostrzeżeń dokonanych na wystawie COMPUTER 70.

ANDRZEJ TARGOWSKI
Krajowe Biuro Informatyki
Warszawa

681.322."313"(73)

Dynamika skali i kierunków rozwoju informatyki w USA

Część I

Artykuł zawiera wrażenia z pobytu delegacji Towarzystwa Naukowego Organizacji i Kierownictwa (na zaproszenie CIMS) w USA.

Dane ogólne o zastosowaniach komputerów

Według informacji uzyskanych w firmie konsultacyjnej MCKISEY, przodującej w USA, przemysł amerykański wyprodukował w 1970 r. 18 tys. komputerów, a łączna liczba eksploatowanych maszyn wyniosła 90 tys. Jest to 2,5-krotny wzrost w stosunku do 1965 r. Podobne tempo wzrostu przewiduje się do 1975 r. — 40 tys. sztuk/rok; 150 tys. sztuk łącznie w eksploatacji.

Łącznie wydatki w sferze zastosowań wynoszą w mld dolarów:

	1965	1970	1975
nakłady inwestycyjne	4,5	12,5	27,5
koszty eksploatacyjne (bez amortyzacji)	4,0	13,0	24,0
Łącznie	8,5	25,5	51,5

Liczba informatyków zatrudnionych w USA wyniosła w 1970 r. 0,5 mln osób wobec 0,15 mln w 1965 r. i przewidywanego 0,775 mln — w 1975 r.

c.d. na III str. okł.

PROJEKT NORMY POJĘĆ PODSTAWOWYCH

Część II – Programowanie (ang. programming)

„Przetwarzanie danych — komputery. Nazwy i określenia podstawowe"

Lp.	Nazwa	Określenie
1	2	3
1	ADRES address	symbol lub numer identyfikujący rejestr, komórkę pamięci lub urządzenie.
2	ADRES BEZPOŚREDNI direct address	adres wyznaczający komórkę pamięci, której zawartość ma być traktowana jako argument operacji.
3	ADRES BEZWZGLĘDNY, ADRES RZECZYWISTY, ADRES MASZYNY absolute address, machine address	1. adres przyporządkowany w sposób trwały przez konstruktora komputera cyfrowego określonej komórce pamięci. 2. część adresowa rozkazu, która bez dalszych modyfikacji określa komórkę pamięci zawierającą argument rozkazu.
4	ADRES EFEKTYWNY ROZKAZU effective address of instruction	adres lub argument rozkazu, uzyskany przez przekształcenie adresu pierwotnego w sposób jednoznacznie określony dla konkretnej maszyny i języka programowania
5	ADRES GENEROWANY generated address	liczba lub symbol, który został wygenerowany w programie za pomocą rozkazów i jest następnie używany jako część adresowa.
6	ADRES POCZĄTKOWY, ADRES PODSTAWOWY base address	liczba używana jako odniesienie przy obliczaniu adresu podczas wykonywania programu.
7	ADRES POŚREDNI indirect address	adres wyznaczający komórkę pamięci, której zawartość ma być traktowana jako adres argumentu, lecz niekoniecznie jako jego adres bezpośredni.
8	ADRES ROZKAZU instruction address	adres miejsca pamięci, w którym jest zapamiętane słowo rozkazowe.
9	ADRES SYMBOLICZNY symbolic address	adres wyrażony w symbolice konkretnego języka programowania.
10	ADRES WZGLĘDNY relative address	liczba, która określa różnicę pomiędzy adresem rzeczywistym a adresem początkowym.
11	ADRESOWANIE addressing	określanie za pomocą adresu części pamięci maszyny albo też innego źródła lub miejsca przeznaczenia danych.
12	ADRESOWANIE BEZWZGLĘDNE absolute addressing	adresowanie za pomocą adresu bezwzględnego.
13	ADRESOWANIE POŚREDNIE indirect addressing	adresowanie za pomocą adresu pośredniego.
14	ADRESOWANIE SYMBOLICZNE symbolic addressing	adresowanie za pomocą adresu symbolicznego.
15	ADRESOWANIE WZGLĘDNE relative addressing	adresowanie za pomocą adresu względnego.
16	ALGORYTM algorithm	sekwencja dokładnie zdefiniowanych operacji lub metod postępowania podejmowanych w celu rozwiązania określonego zagadnienia.
17	ARGUMENT operand	element danych, na którym jest wykonywana operacja.
18	AUTOMATYCZNE PROGRAMOWANIE automatic programming	użycie komputera dla wykonania niektórych etapów czynności przygotowania programu. W szczególności odnosi się to do przetłumaczenia programu napisanego w języku proceduralnym na program w języku wewnętrznym maszyny lub w języku ukierunkowanym maszynowo.
19	BIBLIOTEKA PROGRAMÓW program library	zbiór sprawdzonych programów powszechnego użytku dla określonego typu maszyny, zaopatrzonych w opisy umożliwiające łatwe korzystanie z tych programów.
20	CZĘŚĆ ADRESOWA ROZKAZU address part	część rozkazu częściowo lub całkowicie określająca adres (adresy).
21	CZĘŚĆ OPERACYJNA ROZKAZU operation part	część rozkazu określająca rodzaj wykonywanej operacji.
22	DEKLARACJA declaration	instrukcja języka źródłowego określająca typy zmiennych, obszary w pamięci lub stałe programu.
23	DYNAMICZNY PRZYDZIAŁ PAMIĘCI dynamic storage allocation	technika automatycznego przydziału pamięci, polegająca na rozmieszczaniu w niej programu i danych w sposób zdeterminowany przez warunki powstające w momencie powstania zapotrzebowania wywołanego działaniem programu.
24	DYREKTYWA directive	instrukcja mająca ogólną postać rozkazu programu źródłowego lecz służąca do sterowania tłumaczeniem programu.
25	ELIMINACJA ZER zero suppression	automatyczne usunięcie zer nie znaczących z zapisu liczby.
26	ETYKIETA label	symbol identyfikujący element, zapis lub rozkaz.
27	IDENTYFIKATOR identifier	symbol określający nazwę danych w języku programowania.
28	INSTRUKCJA statement	wyrażenie, deklaracja lub rozkaz języka źródłowego.
29	JĘZYK ADRESÓW SYMBOLICZNYCH, JĘZYK SYMBOLICZNY assembly language	język programowania ukierunkowany maszynowo. Adresy i nazwy danych można wyrażać za pomocą znaków alfanumerycznych, dobieranych przez programistę w sposób dopuszczalny przez dany język.
30	JĘZYK ALGORYTMICZNY algorithmic language	język programowania przystosowany do zapisywania algorytmów.
31	JĘZYK MASZYNY, JĘZYK WEWNĘTRZNY computer language, machine language	język programowania, którego instrukcje są wyłącznie rozkazami maszyny.

1	2	3
32	JĘZYK NIEZALEŻNY OD MASZYNY computer independent language	język programowania, który nie jest językiem ukierunkowanym maszynowo, lecz który w normalnej praktyce wymaga tłumaczenia na różne języki komputerów cyfrowych.
33	JĘZYK PROBLEMOWY problem-oriented language	język programowania przeznaczony do wygodnego formułowania określonej klasy problemów, rozwiązywanych za pomocą komputera cyfrowego.
34	JĘZYK PROCEDURALNY procedure-oriented language	język programowania niezależny od maszyny, przystosowany do zapisywania procesów obliczeniowych za pomocą określania procedur.
35	JĘZYK PROGRAMOWANIA programming language	język przystosowany do pisania programów dla komputerów cyfrowych.
36	JĘZYK UKIERUNKOWANY MASZYNOWO computer-oriented language, computer dependent language	język programowania zbliżony do języka maszyny; jedna instrukcja w tym języku na ogół odpowiada jednemu rozkazowi maszyny.
37	JĘZYK ŹRÓDŁOWY source language	język programowania, którym posługuje się programista; programy w języku źródłowym wymagają tłumaczenia przez translator na programy wynikowe w celu wykonania przez komputer cyfrowy.
38	KOD MASZYNY, KOD WEWNĘTRZNY MASZYNY computer instruction code, machine code	kod użyty do reprezentacji rozkazów maszyny.
39	KOD ROZKAZÓW instruction code	kod użyty do reprezentacji rozkazów języka programowania.
40	KOMENTARZ comment, annotation	tekst słowny wprowadzony do treści programu, wyjaśniający działanie bardziej złożonych fragmentów i tym samym poprawiający jego czytelność.
41	LISTA ROZKAZÓW instruction set	zbiór wszystkich rozkazów, ustalonych dla określonego języka programowania.
42	ŁĄCZNIK connector	1. symbol w sieci działań programu, stosowany w miejscach łączenia kilku dróg w jedną lub rozwidlenia jednej drogi na wiele dróg, 2. sekwencja rozkazów umożliwiająca włączenie do opracowywanego programu istniejących programów lub podprogramów standardowych.
43	MAKROROZKAZ macroinstruction	rozkaz napisany w języku symbolicznym, przeznaczony do realizacji złożonej operacji za pomocą sekwencji rozkazów maszyny.
44	MODYFIKACJA ADRESU address modification	operacja, w której zostaje zmieniona tylko część adresowa rozkazu.
45	NADMIAR overflow	1. stan maszyny spowodowany powstaniem wyniku, który jest większy od zakresu ustalonej reprezentacji liczby, 2. ta część wyniku operacji, która przekracza pojemność komórki pamięci.
46	OPERACJA operation	czynność określona za pomocą pojedynczego rozkazu.
47	OPERACJA ARYTMETYCZNA arithmetic operation	operacja, której argumenty są traktowane jako liczby.
48	OPERACJA LOGICZNA logical operation, logic operation	operacja, której argumenty są traktowane jako ciągi wartości logicznych.
49	OPROGRAMOWANIE software	całokształt środków programowania związanych z określonym modelem komputera cyfrowego. Obejmuje języki programowania i ich translatory, systemy operacyjne maszyny oraz wszystkie programy i podprogramy zawarte w bibliotece programów.
50	PĘTLA loop	sekwencja rozkazów, które mogą być wykonywane w sposób powtarzalny aż do chwili spełnienia określonego warunku.
51	PODPROGRAM subroutine	wydzielony ciąg rozkazów, który może być włączony do różnych programów jako ich część składowa, lub do którego można się odwoływać z różnych miejsc innego programu.
52	PORÓWNANIE LOGICZNE logical comparison	porównywanie dwóch elementów lub pozycji danych w celu stwierdzenia czy są one identyczne.
53	PORÓWNYWAĆ compare (to)	badać dwa elementy lub dwie pozycje danych w celu stwierdzenia ich względnej wielkości, ich wzajemnego umiejscowienia w kolejności lub ich identyczności z punktu widzenia określonych kryteriów.
54	PROCEDURA procedure	reprezentacja algorytmu w języku programowania.
55	PROGRAM program	sekwencja rozkazów w określonym języku programowania umożliwiająca wykonanie przez maszynę operacji niezbędnych do uzyskania z danych żądanego wyniku.
56	PROGRAM BIBLIOTECZNY library program	program wchodzący w skład biblioteki programów.
57	PROGRAM DIAGNOSTYCZNY diagnostic program	program przeznaczony do wykrywania i lokalizowania uszkodzeń w urządzeniach komputera cyfrowego lub błędów w programie.
58	PROGRAM DYRYGENT, DYRYGENT, PROGRAM NADZORCZY supervisory program, executive program	program przeznaczony do sterowania eksploatacji innych programów oraz przebiegu pracy komputera. Program taki stanowi zwykle część systemu operacyjnego.
59	PROGRAM GENERUJĄCY, GENERATOR generating program, generator	program maszyny przeznaczony do tworzenia innych programów przeznaczonych dla wykonywania określonego typu standardowych operacji, takich jak np. programy sortowania lub programy wyjścia.
60	PROGRAM INTERPRETUJĄCY, INTERPRETER interpretive program, interpreter	program tłumaczący przeznaczony do realizacji innego programu przez tłumaczenie każdego rozkazu języka źródłowego na sekwencje rozkazów maszyny i natychmiastowe ich wykonanie. Program interpretujący nie przekształca programu źródłowego na program wynikowy.
61	PROGRAM KOMPILUJĄCY, KOMPILATOR compiling program, compiler	program tłumaczący przeznaczony do przekształcania programów w języku proceduralnym albo w języku problemowym na równoważne programy wynikowe w języku ukierunkowanym maszynowo.
62	PROGRAM MASZYNY computer program	program w języku maszyny.
63	PROGRAM POST MORTEM post-mortem program	program diagnostyczny, umożliwiający wypisanie z pamięci maszyny innego uruchamianego programu.
64	PROGRAM SORTOWANIA sorting program	program przeznaczony do uporządkowania danych wg określonych reguł.
65	PROGRAM SPRAWDZAJĄCY, PROGRAM TESTUJĄCY checking program, tracing program	program przeznaczony do badania innego programu lub danych pod kątem występowania błędów formalnych lub merytorycznych.
66	PROGRAM (PODPROGRAM) STANDARDOWY routine (subroutine)	określona sekwencja rozkazów przeznaczona do rozwiązywania typowych, często spotykanych problemów.

1	2	3
67	PROGRAM SYMULUJĄCY simulator program	program interpretujący, który umożliwia eksploatację programów napisanych dla jednego typu komputera za pomocą maszyny innego typu.
68	PROGRAM TŁUMACZĄCY, TRANSLATOR translating program, translator	program umożliwiający zamianę programu napisanego w jednym języku programowania (źródłowym) na programy napisane w innym języku (wynikowym).
69	PROGRAM USŁUGOWY service program, utility program	program, który spełnia funkcje pomocnicze.
70	PROGRAM WYNIKOWY object program	program, który został przetłumaczony z programu źródłowego.
71	PROGRAM ŹRÓDŁOWY source program	program napisany w języku źródłowym.
72	PROGRAMOWAĆ program (to)	opracowywać koncepcję, pisać i sprawdzać program.
73	PRZESUNIĘCIE shift	przemieszczenie w pamięci komputera bitów, znaków, słów, pól lub bloków danych o równą, określoną liczbę miejsc w wyniku działania rozkazu lub programu.
74	PRZESUNIĘCIE ARYTMETYCZNE arithmetic shift	pomnożenie lub podzielenie liczby przez drugą liczbę będącą całkowitą potęgą podstawy systemu liczbowego.
75	PRZESUNIĘCIE CYKLICZNE cyclic shift	przesunięcie, w którym bity lub znaki przemieszczane z jednego końca słowa lub rejestru są wprowadzane na drugi koniec.
76	PRZYDZIAŁ PAMIĘCI storage allocation	przyporządkowanie miejsc pamięci poszczególnym programom i (lub) danym.
77	PUNKT KONTROLNY checkpoint	miejsce w programie, w którym następuje zbadanie wyników uprzednio wykonanych przez ten program operacji kontrolnych, warunkujących kontynuowanie działania programu.
78	PUNKT ROZGAŁĘZIENIA branchpoint	miejsce w programie, w którym następuje wybór jednej z alternatywnych sekwencji rozkazów.
79	PUNKT PRZERWANIA breakpoint	miejsce w programie, w którym jego działanie może zostać przerwane za pomocą ręcznej interwencji operatora maszyny lub programu nadzorczego np. celem zbadania określonych warunków lub wskazania aktualnego stanu realizacji programu.
80	PUNKT WZNOWIENIA restart point	miejsce w programie, od którego można kontynuować jego działanie w przypadku uprzedniego przerwania.
81	REDAGOWAĆ edit (to)	przygotowywać dane do późniejszej operacji, najczęściej operacji wydawnictwa wyników.
82	ROZGAŁĘZIENIE branch	wybranie jednej z szeregu alternatywnych sekwencji rozkazów, dokonywane podczas każdego przejścia przez punkt rozgałęzienia.
83	ROZKAZ instruction	ciąg znaków, który określa częściowo lub całkowicie operację. Ciąg taki może być użyty w połączeniu z innymi danymi do realizacji żądanej operacji.
84	ROZKAZ ARYTMETYCZNY arithmetical instruction, arithmetic instruction	rozkaz, w którym część operacyjna dotyczy operacji arytmetycznej.
85	ROZKAZ LOGICZNY logical instruction, logic instruction	rozkaz, w którym część operacyjna dotyczy operacji logicznej.
86	ROZKAZ MASZYNY computer instruction, machine instruction	rozkaz określający operację maszyny, która jest realizowana w sposób zgodny z konstrukcją maszyny.
87	ROZKAZ „NIC NIE RÓB" no operation instruction	rozkaz, który nakazuje maszynie nie wykonywać żadnych operacji z wyjątkiem czynności przejścia do następnego rozkazu.
88	ROZKAZ PEŁNY complete instruction, absolute instruction	rozkaz maszyny, który określa całkowicie operację maszyny i który może spowodować realizację tej operacji.
89	ROZKAZ SKOKU jump instruction	rozkaz, który powoduje skok, tzn. przerwanie wykonywanego ciągu rozkazów i przejście do wskazanego punktu programu.
90	ROZKAZ SKOKU BEZWARUNKOWEGO unconditional jump instruction	rozkaz, który zawsze powoduje skok.
91	ROZKAZ SKOKU WARUNKOWEGO conditional jump instruction	rozkaz, który powoduje skok tylko wtedy, gdy zostanie spełniony określony warunek.
92	SCHEMAT BLOKOWY block diagram	schemat graficzny systemu przetwarzania danych lub programu, w którym zadania, czynności i powiązania funkcjonalne są reprezentowane przez umowne symbole graficzne oraz teksty, umożliwiające łatwe zrozumienie ogólnej koncepcji rozwiązania systemu lub programu.
93	SEGMENT segment	wydzielona część programu.
94	SEGMENTOWAĆ segment (to)	dzielić program na części w taki sposób, aby można było go eksploatować bez konieczności umieszczania go w całości w pamięci operacyjnej.
95	SIEĆ DZIAŁAŃ PROGRAMU programming flowchart	schemat graficzny obrazujący szczegółowo rodzaj i kolejność operacji w programie.
96	SKOK jump	odejście w programie od normalnej kolejności wykonywania rozkazów.
97	SKOK BEZWARUNKOWY unconditional jump	skok wywołany działaniem rozkazu skoku bezwarunkowego.
98	SKOK WARUNKOWY conditional jump	skok wywołany działaniem rozkazu skoku warunkowego.
99	SŁOWO ROZKAZOWE instruction word	część lub całe słowo wykorzystywane przez komputer cyfrowy jako rozkaz.
100	SŁOWO ZASTRZEŻONE reserved word	wyraz w języku źródłowym, którego znaczenie jest ustalone zgodnie z określonymi regułami tego języka i nie może być zmieniane w celu dostosowania do potrzeb konkretnego programu pisanego w tym języku źródłowym.
101	STRUKTURA ROZKAZU instruction format	rozmieszczenie w słowie rozkazowym ustalonych poszczególnych elementów składowych rozkazu, np. części adresowej oraz części operacyjnej.
102	SYSTEM OPERACYJNY operating system	program lub zespół programów, który steruje wprowadzanie i wykonywanie innych programów, pracę urządzeń wejścia/wyjścia, tłumaczenie, przydzielanie pamięci i administrowanie danymi oraz umożliwia wykrywanie błędów maszyny i komunikowanie się z operatorem.
103	SYSTEM PROGRAMOWANIA programming system	określony zespół reguł programowania będący łącznikiem pomiędzy określonym typem komputera a jego użytkownikiem, który obejmuje podstawowe programy oraz metody operowania, niezbędne do wykorzystania innych programów.
104	TESTOWANIE PROGRAMU program testing	próbne eksploatowanie uruchomionego programu w celu wykrycia istniejących jeszcze w tym programie błędów.
105	TŁUMACZYĆ translate (to)	przekształcać programy z jednego języka programowania na inny bez zmiany ich znaczenia.

1	2	3
106	URUCHAMIANIE PROGRAMU debug (to)	sprawdzanie poprawności działania programu oraz kolejne usuwanie wykrytych w nim błędów.
107	WYMIENNOŚĆ OPROGRAMOWANIA software compatibility	cecha charakterystyczna oprogramowania komputerów cyfrowych polegająca na możliwości eksploatowania programów napisanych dla jednego modelu na innym modelu komputera. Wymienność oprogramowania może być pełna (dotyczy wszystkich elementów oprogramowania) lub częściowa (dotyczy tylko niektórych elementów oprogramowania, np. języków programowania wyższego rzędu).
108	WYNIK result	skutek wykonania operacji na jednym lub większej liczbie argumentów.
109	ZWROTNICA switch	symbol służący do określenia rozgałęzienia lub sekwencja rozkazów warunkująca wykonanie skoku.

Opracował
Władysław Klepacz

c.d. ze str. 21

W latach 1965—70 dokonano dużego wysiłku w przygotowaniu kadr. W zasadzie potrzeby zostały zaspokojone, a nawet dają się zauważyć pewne nadwyżki, m.in. powodowane redukcjami w programie kosmicznym. Szkoleniem zajmują się producenci (sama firma IBM szkoli 100 tys./rok osób z kadry kierowniczej), prywatne ośrodki i uniwersytety. Producenci zastosowali wobec uniwersytetów długofalową politykę, sprzedając im sprzęt po cenie 60% niższej od powszechnie stosowanej. Dzięki temu uczelnie dysponują najnowocześniejszym sprzętem komputerowym, a studenci nie mają żadnych ograniczeń w jego użytkowaniu.

Dla przykładu Politechnika MIT koło Bostonu dysponuje kilkunastoma komputerami, w tym komputerem najwyższej mocy GE 635 i IBM 360/67. Producenci sprzętu odczuwają już oznaki tej polityki.

Zastosowania w dużych firmach

Analizując poziom zastosowań komputerów w gospodarce amerykańskiej, należy rozróżnić wielkość firm. 500 największych firm wydaje na eksploatację ośrodków obliczeniowych rocznie około 0,25÷5% wartości swojej rocznej sprzedaży, tzn. rząd wydatków wynosi około kilkunastu do kilkudziesięciu mln dolarów. Przykładowo, koncern INTERNATIONAL PAPER COMPANY (odpowiednik naszego przemysłu papierniczgo) zatrudnia 50 tys. pracowników, w tym w informatyce — 550 osób. Wartość rocznej sprzedaży wynosi 1,7 mld dolarów, a wydatki na ETO wynoszą 17 mln dol. (1%). Firma użytkuje 47 komputerów i rozporządza siecią transmisji danych, rozbudowaną w całych Stanach. Do chwili obecnej zainstalowano 156 końcówek komputerowych. Istnieje skomputeryzowany system planowania strategicznego i operacyjnego. Co pół roku modyfikuje się plan roczny na tle 5-letniego planu, który nosi nazwę planu operacyjnego.

Planowanie strategiczne jest prowadzone z wyprzedzeniem 50-letnim, ponieważ tyle czasu trwa cykl odzyskiwania zasobów (firma musi dbać o zasiew drzew). Planowanie strategiczne prowadzone jest głównie z punktu widzenia finansowego, tzn. określa się ryzyko inwestycji, okres zwrotu nakładów oraz warianty lepszego wykorzystania dysponowanych środków finansowych.

Jako ciekawostkę można podać, że bardzo szeroko stosuje się ETO w 5-letnim przewidywaniu zbytu, natomiast w przewidywaniu 1-rocznym raczej technika ta nie jest stosowana ze względu na pewne obawy, czy da lepsze rozeznanie od aktualnego. Celem informatyki w tej firmie jest: minimalizowanie kosztu produkcji, maksymalizowanie zysku i przewidywanie konsekwencji finansowych podejmowanych decyzji.

Jeden z największych amerykańskich banków — BANKERS TRUST COMPANY (prowadzi wymianę z naszym Bankiem Handlowym) — zatrudnia 12 tys. osób i przeznacza 5% swoich kosztów na informatykę (tzn. około 10 razy więcej niż w przemyśle); w zakresie informatyki pracuje tam około 400 osób.

Codzienne prace urzędnicze zostały w 80÷90% skomputeryzowane. Wyeliminowano 500 urzędników z typowych prac. Mimo, że co roku wartość dolara spada o 3%, koszt utrzymania jednego konta bankowego nie wzrósł. 900 osób w centrali banku zajmuje się tylko badaniem skutków ekonomicznych różnych kierunków inwestowania (kredytowania). Przykładowo, na lata 1971—1975 — bank dysponuje 60 wariantami inwestowania. Stało się to możliwe dzięki zastosowaniu ETO.

Sama firma IBM — największy producent komputerów na świecie — ma wiceprezydenta do spraw informatyki, którego roczny budżet wynosi 30 mln dolarów. W firmie tej, zatrudniającej 240 tys. pracowników (rząd wielkości naszego przemysłu maszynowego), w informatyce „wewnętrznej" pracuje 6 tys. osób.

Każdy projekt komputeryzacji zostaje przyjęty do realizacji o ile przewiduje możliwość zwrotu nakładów w ciągu 5 lat.

Zastosowania w średnich firmach

Średnie firmy, których roczną sprzedaż szacuje się na 15—100 mln dolarów (odpowiedniki ZM im. M. Nowotki lub FSO) stosują komputery w takich systemach, jakie prowadzą do zmniejszenia zatrudnienia urzędników, redukcji zapasów i przyspieszenia obrotu pieniędzy. Przypuszcza się, że jest tylko kwestią czasu, aby firmy te przeszły do etapu bardziej zaawansowanych zastosowań prognostyczno-planistycznych, jakie rozwijają duże firmy. Wartość zainstalowanego sprzętu ETO w typowym przedsiębiorstwie wynosi około 0,6 mln dolarów, a roczne wydatki na utrzymanie ośrodka obliczeniowego wynoszą około 0,26 mln dolarów, z czego płace stanowią 60%. Grupa fachowa na ogół liczy 8—9 osób, w tym 2 analityków, 3—4 programistów i 3 operatorów. Zatrudnienie w nowych ośrodkach jest znacznie większe. Charakterystyczne jest, że tylko 10% firm korzysta z usług ośrodków usługowych.

Badanie przeprowadzone wśród 130 tego typu przedsiębiorstw (przemysł, hurt, detal, budownictwo) wykazało, że tylko 5% firm nie dysponuje własnym sprzętem. Najpopularniejszymi zastosowaniami w 50% firm są: zatrudnienie i płace, analiza wykorzystania kredytów bankowych, oceny realizacji zamówień, prognozowanie sprzedaży i planowanie kosztowo-finansowe. Stosunkowo mniej firm stosuje ETO do harmonogramowania dostaw, bilansowania zdolności produkcyjnej (jest ona zwykle w nadmiarze), planowania nowych wyrobów. W latach 1971—1972 przewiduje się szersze opanowanie i tych zastosowań.

c.d. artykułu w nr 7/71

Dni Prasy Technicznej we Wrocławiu a informatyka

W dniach 4—7 maja 1971 r. odbyły się we Wrocławiu — w ramach **Dni Prasy Technicznej** — spotkania kilkudziesięcioosobowej grupy redaktorów czasopism wydawanych przez WCT NOT z władzami terenowymi województwa wrocławskiego, z przedstawicielami aktywu społeczno-gospodarczego oraz Naczelnej Organizacji Technicznej i — na zakończenie — z I Sekretarzem Komitetu Wojewódzkiego PZPR. Przedstawiciele prasy technicznej zwiedzili niektóre zakłady w województwie wrocławskim — jak kopalnie miedzi w Lubinie, DOLMEL oraz Zakłady Elektronicznej Techniki Obliczeniowej — Wrocław. Zaznajomili się również z niektórymi elementami życia kulturalnego Wrocławia i z zabytkami historycznymi ziemi wrocławskiej, tak bogato nasyconej pięknymi pamiątkami polskiej kultury narodowej.

*

Wśród problemów poruszanych na wymienionych spotkaniach — jednym z bardzo często przewijających się tematów był problem rozwoju przemysłu komputerowego w Polsce i wdrażania systemów informatycznych w gospodarce narodowej.

Sprawom tym wiele miejsca w swoich obszernych wystąpieniach poświęcili zarówno I sekretarz KW PZPR tow. L. Drożdż, jak i wiceprzewodniczący Rady Narodowej m. Wrocławia tow. Gryglaszewski oraz przewodniczący Zarządu OW NOT inż. Pietkiewicz. Tematyka ta była często podejmowana w dyskusjach przez redaktorów niektórych czasopism WCT NOT oraz miejscowych aktywistów.

Zwiedzenie ZETO-Wrocław i wysłuchanie wypowiedzi dyrektora tego ośrodka obliczeniowego, a jednocześnie wiceprzewodniczącego Rady Programowej czasopisma „Informatyka" — mgr J. Trybulskiego — zostało ocenione przez uczestników jako bardzo interesujące. Wizyta ta spełniła pożyteczną rolę i przyczyniła się do bliższego zaznajomienia redaktorów czasopism reprezentujących różne branże — po pierwsze — z organizacją usługowego ośrodka obliczeniowego, współpracującego z kilkudziesięcioma różnymi przedsiębiorstwami i placówkami, i po drugie — z używanym tu sprzętem, m.in. z polskim komputerem do przetwarzania danych ODRA 1304, zainstalowanym tam od ub. roku, i z jednym z pierwszych funkcjonujących w Polsce łączy teletransmisji danych. Wizyta ta również dała w pewnym stopniu poglądową ilustrację niektórych problemów poruszanych generalnie w czasie spotkań z władzami województwa wrocławskiego.

Jak wynikało z wypowiedzi na spotkaniach wrocławskich, jedną z poważnych trosk władz i aktywu regionu wrocławskiego jest przyszłość Zakładów ELWRO i rola ELWRO w rozwoju informatyki polskiej. Rozbudowie zdolności produkcyjnej ELWRO poświęca się tam od lat wiele uwagi. Na terenie Wrocławia rozwija się intensywnie szkolnictwo wyższe i średnie, przygotowuje się kadry dla potrzeb produkcji i obsługi maszyn matematycznych. Zdaniem aktywu wrocławskiego, Zakłady ELWRO mogą podejmować o wiele większe zadania, aniżeli te, które stawiają im obecnie plany Zjednoczenia MERA. Istnieją jednak kontrowersje między środowiskiem wrocławskim a władzami centralnymi co do lokalizacji produkcji poszczególnych wyrobów, co do organizacji obsługi użytkowników, co do wykorzystania zaplecza naukowo-badawczego, co do perspektywicznego rozwoju całej branży. Niełatwe te sprawy są obecnie w toku żmudnego konfrontowania i uzgadniania.

Władze regionu wrocławskiego dużo uwagi poświęcają sprawie wdrażania nowoczesnych metod zarządzania i zastosowań ETO. Na spotkaniach padło wiele krytycznych uwag o działalności koordynacyjnej byłego Urzędu PRETO. Zapowiedziano przygotowanie narady na temat nowych metod zarządzania. Planuje się zorganizowanie w dwóch zakładach przemysłu maszynowego na terenie Dolnego Śląska — rodzajów uniwersytetów w zakresie nowych form zarządzania. Na Politechnice Wrocławskiej powstał specjalny kierunek organizacji przemysłu, przygotowujący projektantów systemów elektronicznego przetwarzania danych. Uniwersytet Wrocławski intensywnie szkoli programistów. Wrocławskie uczelnie i placówki naukowo-badawcze — w tym również Wyższa Szkoła Ekonomiczna — stanowią poważne zaplecze dla informatyki. Rozwija się ZETO-Wrocław, zwiększając stale potencjał obliczeniowy oraz stan zatrudnienia (do końca 1971 r. — 290 osób). ZETO-Wrocław przystępuje obecnie do stosowania racjonalnej polityki wyboru systemów i klientów z punktu widzenia ciężaru gatunkowego w gospodarce narodowej. Rachunek ekonomiczny zaczyna odgrywać coraz większą rolę. W zakresie eksploatacji maszyn czyni się starania o obniżenie cen usług świadczonych przez ZETO. ELWRO obniżyło ceny sprzedaży systemów ODRA. ELWRO-Service przyrzeka do końca br. zlikwidować wszystkie zaległości w zakresie instalowania u użytkowników wyprodukowanych maszyn (co jednak nie załatwia otwartego dotychczas problemu generalnego dostawcy systemów oraz terminowości napraw gwarancyjnych i pogwarancyjnych).

Posunięcia te powinny wpłynąć stymulująco na wzrost zastosowań ETO w gospodarce, a jednocześnie — na usprawnianie zarządzania.

Powyżej zasygnalizowano tylko, w prawie telegraficznym skrócie, problemy dotyczące informatyki, a wybrane z całej bogatej treści spotkań wrocławskich, w czasie których redaktorzy prasy technicznej mieli możliwość szerokiego zaznajomienia się z osiągnięciami (a także z bolączkami) Dolnego Śląska — jednego z najsilniej rozwiniętych gospodarczo regionów Polski.

Trudne problemy rozwoju informatyki w Polsce są przedmiotem licznych dyskusji środowiskowych, a więc powinny być w dalszym ciągu gruntownie i kompetentnie przedstawiane na łamach czasopisma „Informatyka" w postaci odrębnych artykułów i wypowiedzi, o których nadsyłanie Redakcja bardzo prosi.

D. P.